電腦時代

COMPUTING AGE

3

微型電腦用家雜誌

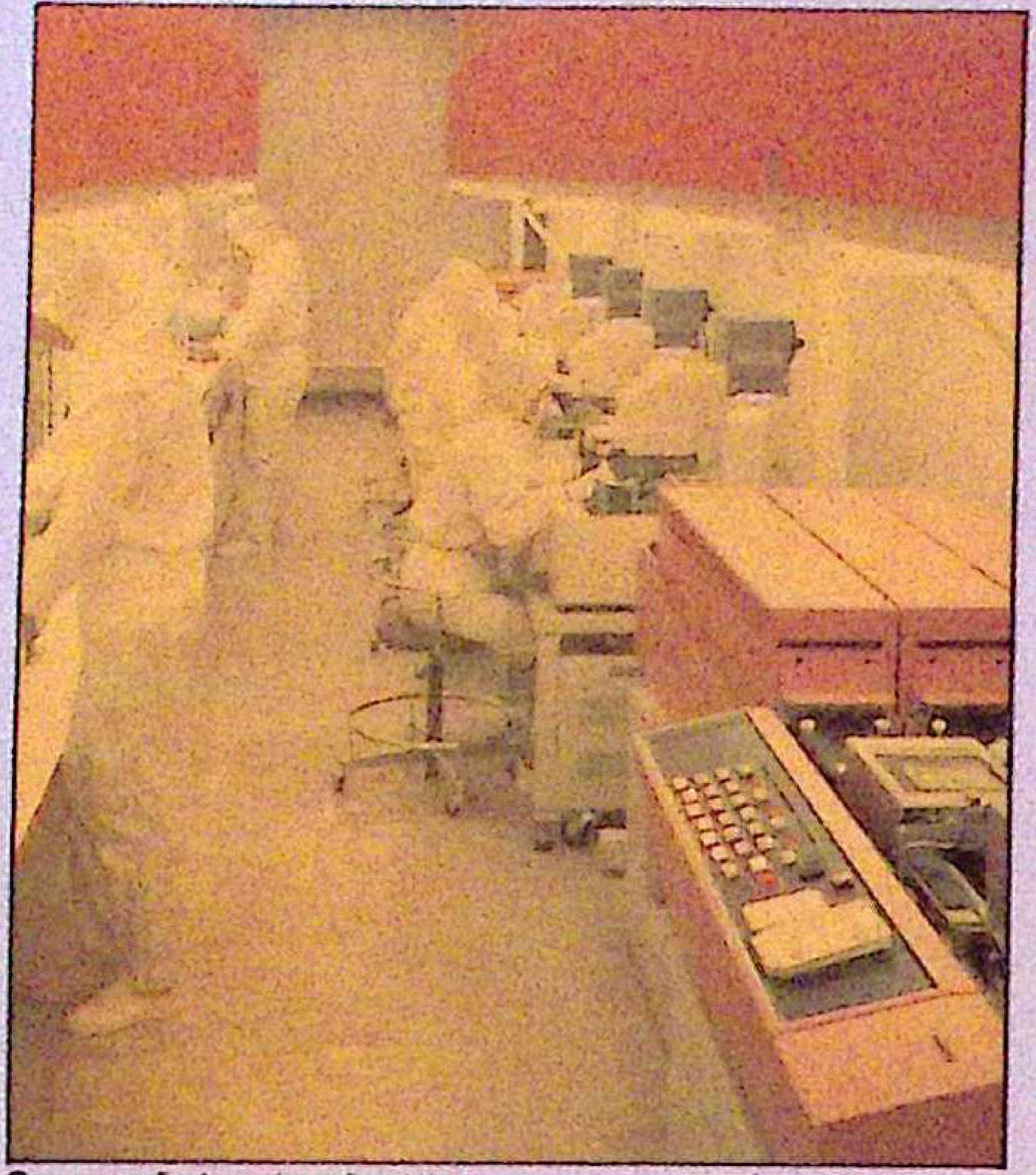
HK$8.00

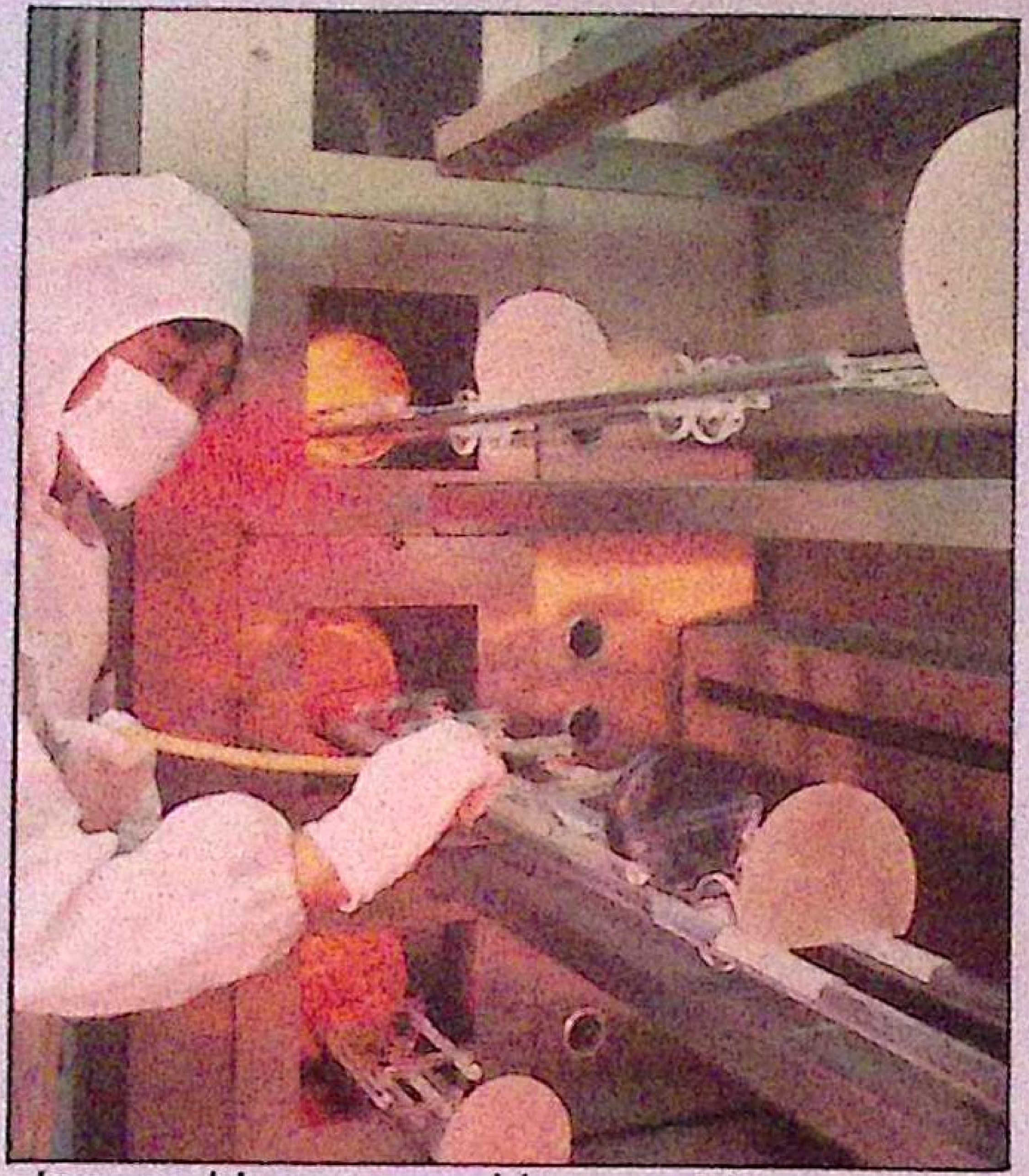


- 神經病醫生
- 磁碟目錄封套
- APPLE 音樂特輯
- APPLE PASCAL



Our latest CAD facilities are just one indication of our strong commitment to customer design services.





State-of-the-Art facilities and state-of-the-Art technology enable us to provide you with complete silicon-foundry services. Turn-around cycle is just 20 DAYS.

From A New Leader in Silicon Gate CMOS Technology

Our success has been based on manufacturing circuits on the complimentary MOS (CMOS) technology featuring consistently high performance and reliability. Because of the intrinsic superiority in respect to noise immunity, a wide variety of voltage — supply conditions, thrifty use of power, and temperature insensitivity, the CMOS technology is especially ideal for use in harsh environments where no other technology can withstand.
We carry a standard IC line and regularly provide customer design and foundry services. The turn-around cycle for foundry services can be as short as 20 days.

Product Number	Process	Description	TTL or CMOS Compatible	Pin Compatible with
MICROPROCESSOR, MEMORY AND LOGIC				
HKE65SC02	CMOS	8 bit Microprocessor	TTL and CMOS	NMOS 6502
HKE5114	CMOS	1Kx4 Static RAM	TTL and CMOS	2114 4K SRAM
HKE74HCTxxx	CMOS	High Speed CMOS Logic Family	TTL and CMOS	74 Series TTL
HKE74HCxx	CMOS	High Speed CMOS Logic Family	CMOS	74 Series TTL
WATCH CIRCUITS				
HKE151	CMOS	32 KHZ analog	Watch circuit	MB 151
HKE5001	CMOS	3½ digit 5 function multiplexed LCD		MSM 5001

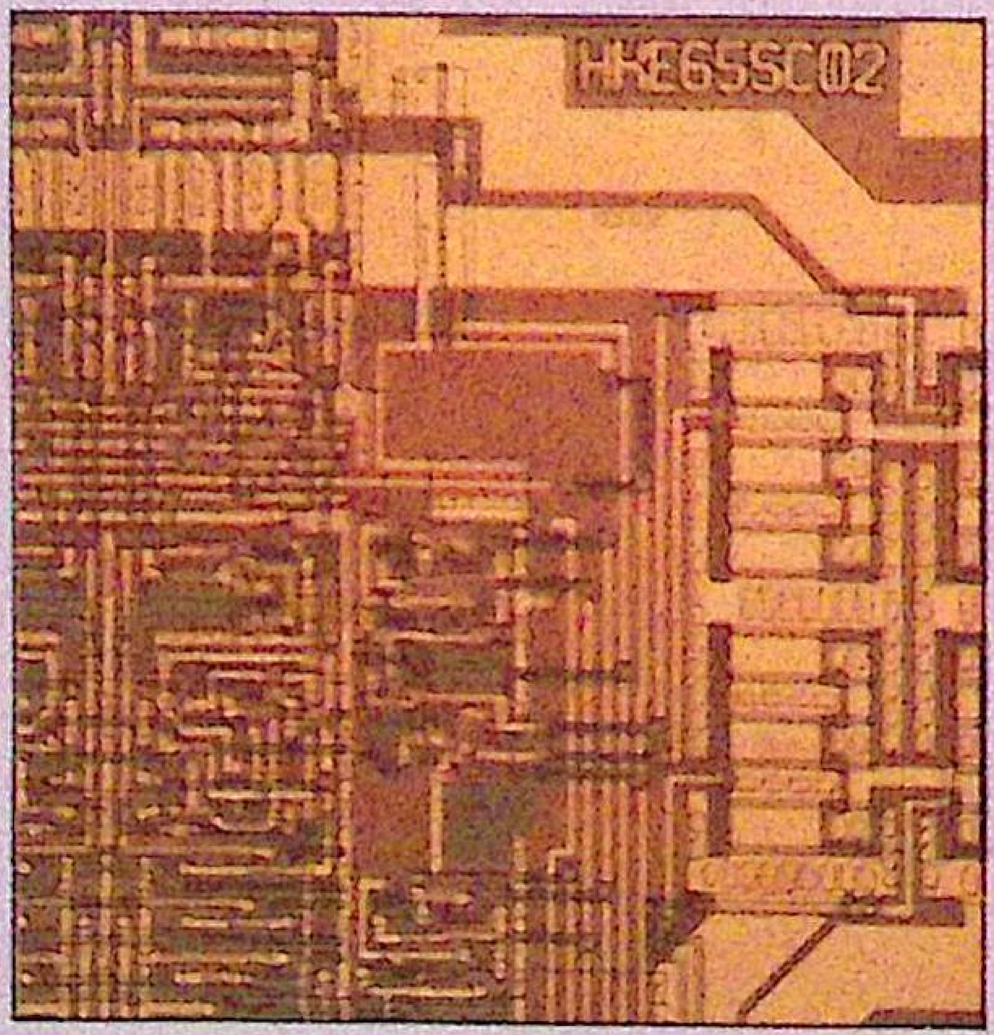
THE SYSTEM DIVISION OF HUA KO
is developing and marketing personal computers and medium size computers for general purpose and turn-key applications.

THE ORGANIC MICRO-modular units for expanding and upgrading Sinclair ZX81 and ZX Spectrum personal computers.

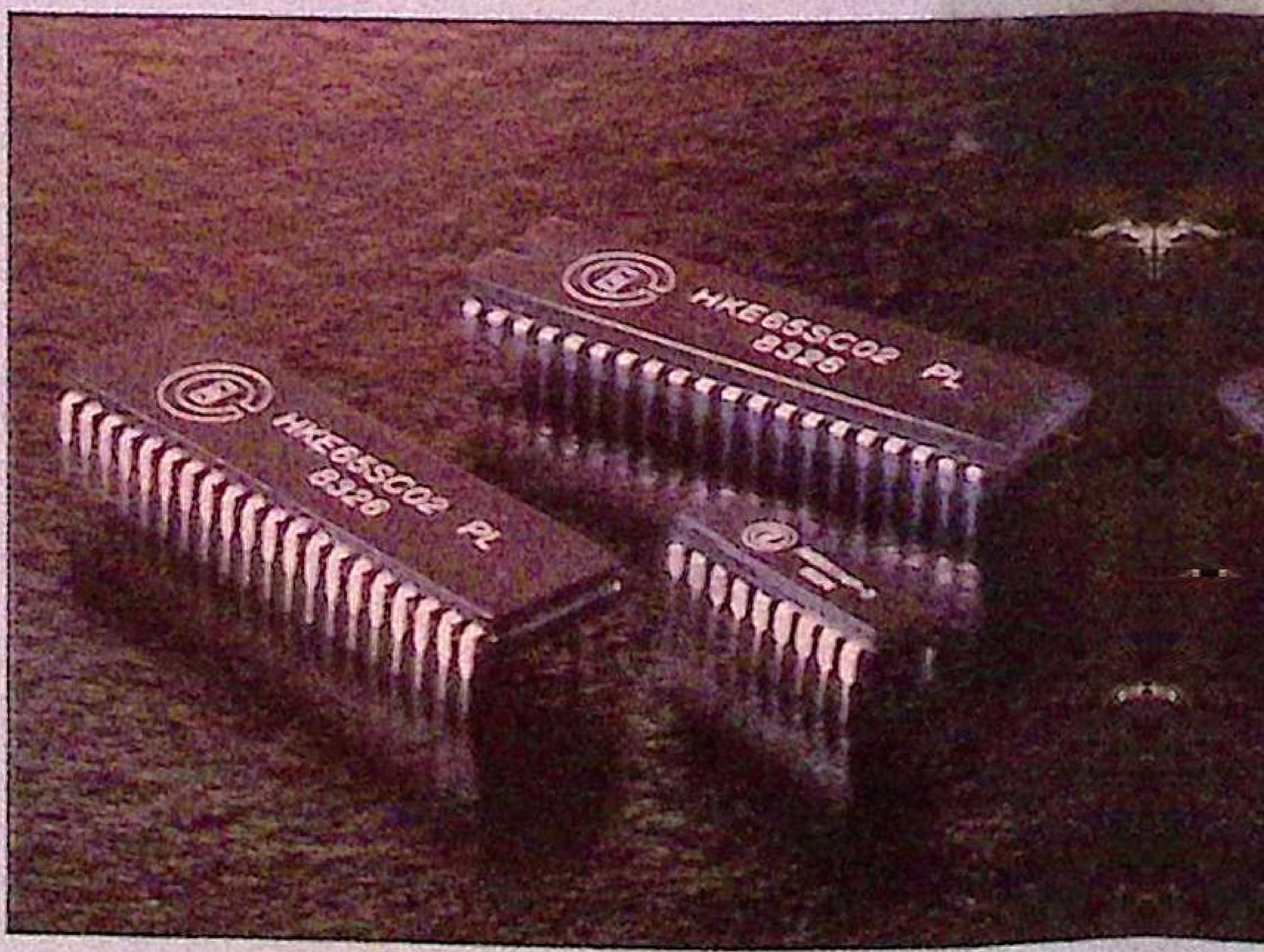
- PERSONA — Interface module enables Organic Mirco to grow on your computer
- MINIMAP — Memory mapping to extend to a maximum of 1M bytes!
- RAM 08 — Low cost memory expandable from 2K to 8K
- RAM 16 — 16K add-on memory at remarkable low cost
- RAM 64 — A TRUE 64K add-on memory
- DROM — For non-volatile storage of programs and data
- TOOLKIT — A module to take 8K bytes of utilities in EPROM/ROM
- PERICON a — Providing 24 lines of input/output
- PERICON b — 24 lines of heavy duty output-access the outside world
- PERICON c — Drive an 80 column printer with Centronics type interface
- SONUS — Three-voice synthesizer, sound generator with stereo option!
- LINK a — An 8 channels A/D Converter

IBM PC compatible Add-on Memory board
- HKE 64-pc — 16K Bytes
- HKE 128-pc — 128K Bytes
- HKE 192-pc — 192K Bytes
- HKE 256-pc — 256K Bytes

IBM is the registered trademark of International Business Machines.



4 micron design rule and ISO-CMOS silicon gate technology are used in the manufacture of the HKE65SC02.



The very first product announced by HUA KO is an 8 bit CMOS microprocessor, the HKE65SC02, pin-to-pin plug-in compatible with the NMOS 6502.

HUA KO ELECTRONICS CO. LTD.
Head Office & Factory: 9, Dai Shun Street, Tai Po Industrial Estate, Tai Po, N.T.,HONG KONG.
TEL: 0-6570373 (10 lines) TELEX: 50898 HKETP HX CABLE: 7599


start
stop

TAXAN彩色電腦螢幕

VISION — EX 領先產品

RGB,COMPOSITE兩種功能融滙一身

內置揚聲器，可配用於任何電腦及錄影機。

- RGB及COMPOSITE兩種系統集于一身，12吋螢幕，眞實顏色顯像，效果更佳。
- 售價更普及，比一般COMPOSITE COLOUR MONITOR更爲相宜。
- COMPOSITE SIGNAL有PAL及NTSC兩種系統可供選擇。
- APPLE或AP電腦只需加上 RGB CARD即可使用RGB系統。若使用COMPOSITE系統更不需另加任何附件。兩種系統，顏色明亮艷麗，字體清晰完美。
- 使用TAXAN RGB CARD II，更可將TEXT及BACKGROUND轉換各種顏色。
- 其他備有RGB SIGNAL之電腦，除接駁電線外，無需加設任何附件便可直接使用。
- 內置音响系統，0.7 W輸出功率。
- 用於電視錄影機時，噪音特低，畫面穩定，顏色鮮艷，層次分明。
- 附有防輻射護眼鏡片，觀看時更加安全舒適。

*所有 **TAXAN RGB VISION** 系列之彩色電腦螢幕，均採用''線性放大影像系統''（**LINEAR AMPLIFIER VIDEO CIRCUIT**）可充份顯示所有顏色組合，絕無缺少顏色現象，功能超越其他牌子。顧客購買同類電腦螢幕時，請測試該螢幕是否在RGB系統下能夠顯現各種不同顏色，以保証品質。

享有代理商保証，才是專業性保用，愼防水貨及二手貨，請索取全新保用証，如証上有任何塗改，即自動作廢。如有疑問，請立即向本公司查詢。

最新抵港



特約經銷商:

Golden Micro Computer Co.
3-793168, 3-602564

Zirius Company
3-667471

Yiu Lee Company
3-790150

總代理

VINCENT COMPUTER CENTRE LTD.
(A Subsidiary of Vincent Group)

雲生電腦有限公司（雲生集團附屬機構）

九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

:RITEMAN: 超級印字機

超值售價　超速印字　超薄機身

特點：

- 機身體積爲普通印字機一半。
- 超速印字，每秒可印120字。
- 雙向式穿梭印字，每分鐘可達63行。
- 多種印字功能，可印四國文字，更可作Graphic Dumping（需另加Graphic Interface）。
- 採用新設計之盒式絲質墨帶，用戶可補充油墨重覆使用，操作簡易，墨帶壽命高達一百萬字。
- 減少更換墨帶次數，不斷節省金錢，物超所值。

長期節省開支，半年慳到差價，二十個月慳到一部新機。

	每盒售價	可印字數	每年印字數	需用墨帶	總開支
普　通　墨　帶	HK$60	600,000字	20,000,000字	33盒	HK$1,980
Riteman可補充油墨墨帶	HK$40	1,000,000字	20,000,000字	20盒	HK$　800

每年節省開支達HK$1,180.

獲美國聯邦通訊委員會，
標準檢定合格，證書編號
CWT9Z6 RITEMAN，
以保証品質。



享有代理商保証，才是專業性保用，愼防水貨及二手貨，請索取全新保用証，如証上有任何塗改，即自動作廢。如有疑問，請立即向本公司

特約經銷商：
Golden Micro Computer Co.
3-793168, 3-602564
Profit's Computer Co.
3-780983
Sunray Electronics Co.
3-7237390

總代理
VINCENT COMPUTER CENTRE LTD.
(A Subsidiary of Vincent Group)
雲生電腦有限公司
(雲生集團附屬機構)
九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0
電訊：HX38316 VCC HK

BMC BM-12ES Monitor

設計更精，質素取勝

特點：

- 配用於一般電腦及文字處理系統，完美無瑕。
- 畫面穩定，字體清晰。
- 12″綠色螢光幕，特備護眼鏡片，舒適安全，久看不倦。
- 外形美觀，可配精美原廠旋轉底座，實用方便。
- 全部日本製造及裝嵌。

SPECIFICATIONS

Input Signals	Composite Video Signal, Negative SYNC. $1.0^{+0.5}_{-0.3}$ Vp-p, 75 ohm
CRT Size	31 cm diag. (12 inch diag.)
Phosphor	P31 (Green)
Video Amp Bandwidth	18 MHz
Display Area	Horizontal 21cm (46 μS) x Vertical 15cm (18.23 mS)
Display Format	1920 Characters max. (80 char. x 24 lines)
Scanning Frequency	Horizontal 15.80 kHz, Vertical 50 Hz
Power Input	AC 230 V, 50 Hz
Power Consumption	26 W
Dimensions	32.0(W) x 27.9(H) x 30.8(D) cm
Weight	7 kg.

* Specifications are subject to change without notice.

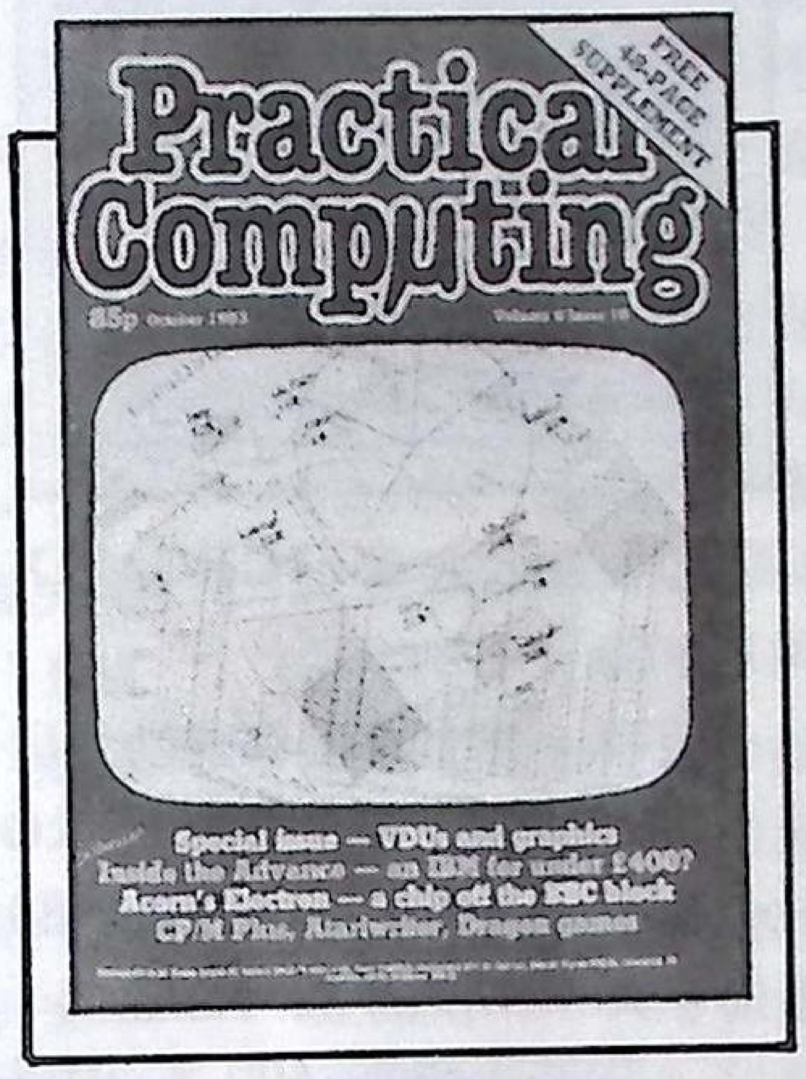


享有代理商保証，才是專業性保用，慎防水貨及二手貨，請索取全新保用証，如証上有任何塗改，即自動作廢。如有疑問，請立即向本公司查詢。

總代理
VINCENT COMPUTER CENTRE LTD.
(A Subsidiary of Vincent Group)
雲生電腦有限公司（雲生集團附屬機構）
九龍尖沙咀柯士甸路22-26號好兆年行608室 電話：3-7211288-0 電訊：HX38316 VCC HK

歐美國家權威電腦雜誌選出

一九八三年最佳電腦



英國權威電腦雜誌
PRACTICLE COMPUTING



美國權威電腦雜誌
PERSONAL COMPUTING



法國的權威電腦雜誌
MICRO 7

歐美國家幾本權威性電腦雜誌聯合進行評選去年（一九八三年）最佳電腦。他們一致評定的結論如下：

最佳家庭電腦： COMMODORE 64
最佳私人電腦： IBM PC
最佳手提套裝電腦：KAYPRO Ⅱ

進行評選的電腦雜誌包括：

英國的 PRACTICAL COMPUTING
美國的 PERSONAL COMPUTING
法國的 MICRO 7
意大利的 BIT
西班牙的 CHIP
荷蘭的 MICROMIX
西德的 CHIP

榮獲最佳家庭電腦的 COMMODORE 64 於一九八二年底開始推出發售，初期僅供應有限數量，售價爲 340 英磅，及至在英國 NORTHAMPTONSHIRE的CORBY正式大量生產，售價減至 229 鎊。它是該年度暢銷家庭電腦。

COMMODORE 64 採用MOS 技術由 6502發展出的6510 爲CPU 。有 64 K RAM。螢幕顯示爲 40 字符×25 行，或 300 乘 400 像素圖案，有十六種色彩。其BASIC沒有 SOUND 指令，但發聲設計非常強效，有九個八音度。

配同 COMMODORE的有一系列的週邊設備，包括磁碟機、印字機、卡式機。它更可接連硬碟記憶體。

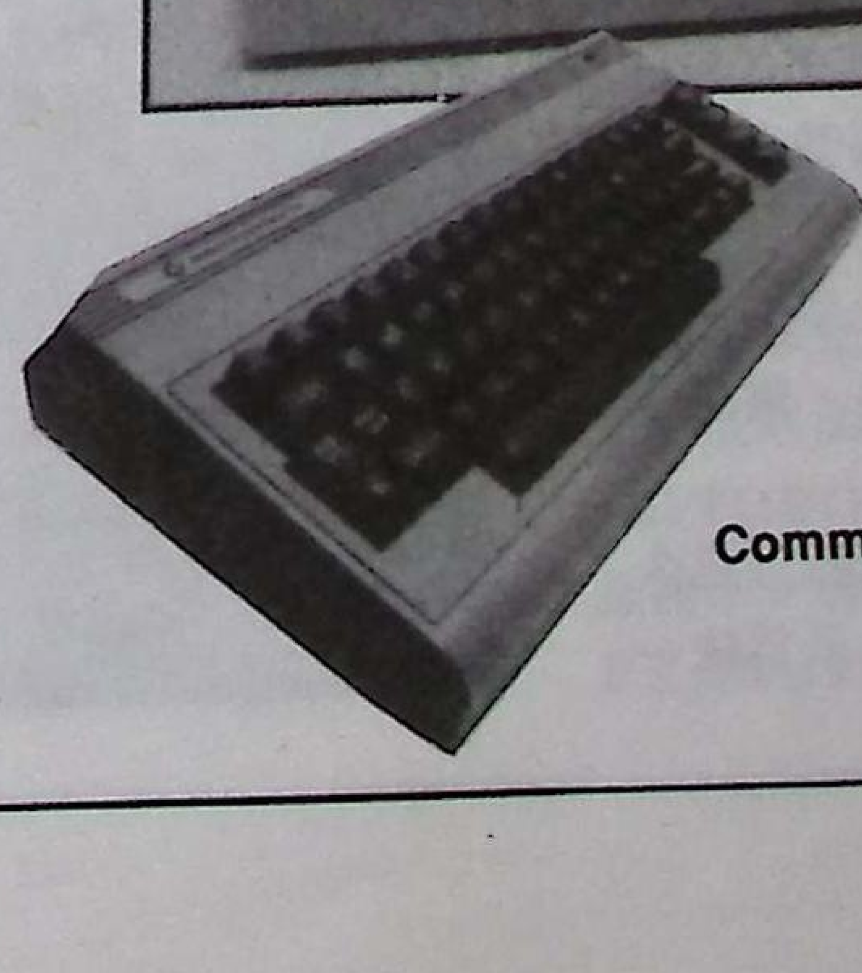
至於獲選成爲一九八三年最受歡迎的私人電腦的 IBM PC 則是衆望所歸。

它在美國推出不久，即能攻佔一個穩固而不斷有增長的市場。更甚的是，導致許多相容或相似的產品湧塞市場。

IBM PC是採用 INTEL 的 8088爲CPU（16 BIT）。RAM 記憶量可由 64 K增加至 512 K 。螢幕



Kaypro II.



Commodore 64



意大利的權威電腦雜誌　BIT



西班牙電腦雜誌　CHIP



荷蘭電腦雜誌　MICROMIX



西德電腦雜誌　CHIP

顯示爲 80 或 40 字符乘 25 行，或 640×200 圖案像素。標準系統有一單音調聲音產生器。

標準系統包括有兩部 360 K 磁碟機，使用PC—DOS 操作系統（此 OS 是由 MICROSOFT 的 MS—DOS 發展而成）。其加強系統 PC XT 有 10M BYTE 的硬碟設備。

有跡像顯示，IBM PC 會成爲商業軟件及執行工具所使用的標準電腦。

手提式套裝電腦 KAYPRO Ⅱ 於一九八二年底推出後不久，即能在美國成功地佔一地位，能獲得最佳手提式套裝電腦實非僥倖，KAYPRO Ⅱ 有著濃厚的 OSBORNE 電腦影子。CPU 是用 Z—80，運用 CP／M。有 64K RA—M 及兩具 5 ¼吋磁碟機，提供的記憶量達到每部 200 K。同時更備有 400 K 的磁碟機版本及 10 M BYTE 的硬版本。附同的 9 吋（對角綫）綠色螢幕可顯示 80 字符×24 行。此機並沒有色彩及聲音設計，但其優異處在於有一系列的軟件支援，例如：CP／M 2.2，PROFIT PLAN，M BASIC 等。□

TAXAN產品推展會 介紹一系列電腦螢幕

雲生電腦有限公司代理日本名廠加賀電子株式會社之 TAXAN 電腦產品已有數年。

該公司爲推廣及介紹 TAXAN 之新產品特在十二月八日假尖沙咀東之富豪酒店凡爾賽殿舉行酒會及介紹展招待各大分銷及零售商。

展出包括最暢銷之KG系列綠色電腦螢幕，REB VISION 系列高解像度彩色電腦螢幕，可適合電視錄機及電腦使用之全新 RGB VISION EX 系列彩色電腦螢幕，全新 KX高解像度綠色電腦螢幕。以上電腦營幕均適合 IBM PC，APPLE Ⅱ 及一般電腦使用，會場更展出可供教學用之電腦控制機械臂，各種先進產品，琳琅滿目。

加賀電子株式會社副總裁四方堂久登先生，更在會上致詞，他表示爲統一世界各地對加賀產品之商標，特在一九八三年四月開始將 KAGA 商標更改爲 TAXAN 。對此項更改而帶來零售商及用戶一些不便之處深表歉意。

他更展望在一九八四年內TAXAN產品在港及中國大陸之銷量將會大增。

當日出席人士包括各大分銷商及零售商，冠蓋雲集，場面熱鬧。



花生鬥蘋果

足以影响大勢的兩間公司

IBM與APPLE各展奇謀爭奪市

家駒

PEANUT（花生）的硬殼終於在上月（十一）一日裂開，期待已久的IBM家庭電腦即將在全球面市，除了大吹大擂引起一陣子刺激及注意外，PEANUT 家庭電腦似乎令人有點失望，除了價格較廉宜，及可以應用 IBM PC電腦的軟件外，PEANUT 並沒有令人驚喜的新東西。

過去一年有關 PEANUT 的傳聞與揣測在電腦界及美國華爾街分析家中，不脛而走，引致不少誤導資料滿天飛。

花生的模樣

但現在 PEANUT 已揭開了它的神秘面罩，新產品淸楚地表露無遺。究竟 PEANUT 是甚麽？PEANUT 能否在家庭電腦市場異軍突起，廣受歡迎，一如 IBM PC 電腦在過去兩年大出風頭，成爲商業電腦市場的典範及標準？PEANUT 是否能在市場上分一杯羹，或者會被「蘋果」廠短期內推出的 MACINTOSH 所壓倒，被這個強勁對手的聲勢所掩蓋過？我們曾經透過可靠消息來源，搜集到一些資料，或者可以對上述問題分析一下。

PEANUT 家庭電腦，售價會在六百美元至七百美元之間，是用 16 BIT INTEL 8088 微處理器，一如IBM PC 所採用者。內有64 K BYTES RAM（可以擴展至 128 K。有鍵盤，5¼吋磁碟驅動器，兩個擴展槽（由於沒有印字機介面，故此其中一個是留給印字機。）及彩色繪圖。PEANUT的外型與IBM PC 非常相似，PEANUT 插入標準的電視機，可以顯示四十欄文字，連同 PC—DOS 2.0及MICROSOFT BASIC 一同出廠，不過 PEANUT 如果要操作PC電腦的程式，就要加設 80—COLUMN的承接器，監視器及印字機，這些配件會令PEANUT的價格推高至一千三百美元至一千四百美元。將 IBM PC 軟件互爲使用的最大問題是 PEANUT 的記憶極限只是 128 K。

由於IBM擁有 INTEL 的大部份股權，所以不少人都猜測究竟 PEANUT 採用那種微處理器，是 8088，抑或是最先進的 80188，如果用 80188，就可以與 IBM PC 互爲使用，而由於減少採用集成電路晶片，生產成本價格亦相應下降。不過，根據消息來源說，在設計階段時PEANUT是採用 80188微處理器，但由於來源供應短缺關係，初期只會用 8088，後期生產才會改用 80188，而原本計劃，利用紅外光或高頻率科技使無綫鍵盤可以與房內電視機連繫，亦受拖延。如果低價格的無綫鍵盤的可靠性獲得改進，肯定會成爲隨意附加的配件，供消費者選擇。

短缺記憶

分析家說，PEANUT 最大倒退的地方，是記憶能力極限。事實上，IBM希望現時在辦公室已擁有IBM PC的行政人員，會購買 PEANUT，它的 PC—DOS 操作系統，可令行政人員帶他們部份的工作回家處理。但問題是，很多工作是需要 192 K至256 K BYTES的記憶容量才能處理較先進的 IBM PC程式，故此 PEANUT 的極限只有128 K BYTES，能否勝任愉快，實屬疑問。

IBM 進軍家庭電腦市場，標誌着家庭電腦不再是玩遊戲的玩具，市場分析家說 PEANUT 的面世會令消費者對家庭電腦的觀念完全改變，並能刺激市場的需求，這個新品種家庭電腦加上 IBM 的聲望，會掀起 PEANUT 熱潮，當這種情況出現，PC—DOS 將會成爲家庭電腦市場的自由標準，一如 MS—DOS（屬於 PC—DOS 系列）變成商業電腦市場的標準一樣。系統基礎穩定後，會刺激很多獨立軟件發展商生產更廣泛系統的電腦軟件，附有更大程度的互爲使用的性能。

蘋果的威脅

IBM PEANUT 可以預料受到蘋果電腦廠的強勁競爭威脅。首先，蘋果 IIE雖然是 8—BIT系統，但第一眼感覺與 PEANUT 並無多大差別。而蘋果 IIE 八四年初推出新的 PRO—DOS操作系統，功能上會大大改進。

此外，蘋果 IIE 隨意附加的硬碟，會令它成爲威力巨大的電腦，並可對任何的微型電腦提供廣泛的軟件支援。

對大多數消費者，最重要的是價錢問題，「蘋果」廠預料利用割價或加送零件，令 IIE更具競爭能力。IIE 的售價會爲五百美元，價格比 PEANUT 爲廉宜，而且有能力改進至 CP／M 操作系統。

強勁的對手？

其次，蘋果最新的電腦品種 MACINTOSH，仍在閉門研究設計中，相信會是 PEANUT 最強勁的對手。雖然 MACINTOSH 在過去兩年不斷改良研究，惟是最後的設計，蘋果總部曾經爭論不休。事實上，消息指出，蘋果廠分開兩個部門分別研究 LISA 及MACINTOSH 計劃，互相競爭下，却令兩者同受阻延，最近預測蘋果會在八四年初推出 MACINTOSH。

不像 PEANUT，標準的 MACINTOSH 包括監視器專業鍵盤及磁碟驅動器。MACINTOSH 的售價預料爲二千美元，MACINTOSH 是麗莎電腦再改良的產品，利用的科技比 PEANUT 更爲先進。

MACINTOSH 與 PEANUT 的比較應該是在功能與價錢上的。兩者的差距是兩面性的。MACINTOSH是用 MOTOROLA 68000，32BIT處理器操作，每次處理16—BIT 資訊，PEANUT像 IBM PC，用 16—

BIT 處理器，每次處理 8－BIT資訊，MACINTOSH 用較新，甚至是較複雜先進的操作系統，以對抗 PE－ANUT PC－DOS 操作系統。

基於它的速度，68000 可以處理複雜的功能如 BIT－MAPPED繪圖，而無需減慢軟件的運作，而這是其他微處理器共同的缺點，擧例，在BIT－MAPPED 繪圖時，你可在監視器看到了圖形一如在紙上出現一樣，並可用比例性的字母及黑體大字加於其上，BIT－MAPPED 系統有能力做出高解像度的繪圖，圖表及圖解。這個特性會為家庭使用者所喜愛及歡迎。

再者，蘋果廠已下注及籌劃擴大 MACINTOSH 應用軟件的範圍基礎，蘋果的總行政員寇利已下令 MAC－INTOSH設計可以與 IBM PC 互爲使用。不久以前，生產一部蘋果電腦配合其他牌子微型電腦是難以置信的，但寇利顯然是想讓 IBM訂出電腦業的標準，而着力凌駕其上。

努力實現相容使用

至於蘋果如何能夠實現與 IBM電腦配合使用仍待觀察，因爲 MACI－NTOSH 所採用的微處理器與 PE－ANUT 不同，要令 MACINT－OSH 與 IBM 電腦互爲使用實在不是易事。蘋果已宣佈，LISA 可使用 MS－DOS ，或者是用額外的 8088 卡插入擴展槽內，不過 MACINTOSH 好像不如 LISA 或 APPLE IIE 一樣，有內在擴展槽。但蘋果廠可以在 68000上做點魔術，使它可以應用寫在其他微處理器上的程式。

MACINTOSH 的幾項特性令它有很大的市場潛力，首先，它不同 P－EANUT，MACINTOSH 是特別爲商業上用途而設計，它較大的記憶容量，及高解像度監視器可以支持達到此目的。其次，MACINTOSH 在教育用途上亦大力推廣，美國賓夕凡尼亞一間大學已簽訂合約購買三千部蘋果電腦，大學已通告所有大學新生（約一千八百人）必須購買一個系統，每個系統均有文字處理及 SPREADSHEET 軟件，第三，過去，過去十八個月，幾家獨立軟件發展商已爲 MACINTOSH 做好準備，當它面世後，即有大量軟件出現支持 MACINTOSH 。

由於 MACINTOSH 及 PEAN－UT 價格較高，故此市場分析家認爲將會令家庭電腦使用者劃分爲兩類，一類會買價錢較貴的 IBM或蘋果電腦，另一階層則會購買價錢較平的 ADAM，COMMODORE 64 等家庭電腦，在後一型電腦， COMMODORE 顯然會處於最有利位置，因爲它鍵盤加中央處理器，沒有磁碟驅動器及印字機，售價僅二百美元， COMMODORE 廠因有能力生產完整電路晶片，成爲市場上最佳武器而預測在未來十二個月內，C－OMMODORE 64 家庭電腦銷量會衝破二百萬大關。

其他公司變應辦法

COLECO ADAM 是走 PE－ANUT 與 COMMODORE 64的中間路綫，它與 COMMODORE 64 一樣，價格非常廉宜，而與 PEANUT 一樣，可以擴大記憶容量， ADAM 是眞正的"完整"系統，只要插入之後，就可以開始輸入並進行文字處理工作。

但 ADAM 亦有其缺點，它的印字機操作是歷來電腦印字機最慢者，傳言說是因墨帶驅動效能欠佳，評論者並懷疑 COLECO，這家電視遊戲製造商，是否能夠在八三年達到五十萬部銷售目標。不過，在十二個月內，可以應用 CP／M的 ADAM，銷量預料會比 PEANUT 多兩倍，而COMMODOR－E 64 爲了達到銷售二百萬部目標，部分機器亦會改裝，可以應用CP／M，預料 CP／M 會對家庭電腦獨立軟件製造商有最大吸引力，而非MS－DOS。

TEXAS INSTRUMENTS 及 ATARI 兩家電腦廠情況欠佳，家庭電腦銷售緩慢令兩家廠處於惡劣地位，而與 COMMODORE 進行的割價戰，令他們損失億計美元。TAXAS IN－STRUMENT 於上月亦正式宣佈終止TI99／4A的生產。ATARI能否回復過去叱咤風雲的光景，則要看它重新研究改良新系列家庭電腦的表現了。

日本的入侵

上述所說的因素，造成日本公司期待已久時刻的來臨，MS－DOS 發展商，MICROSOFT ，已向廿五家日本製造商發出許可證，日本已有開拓美國市場之匙，因爲MS－DOS 操作系統已廣受接納，日本公司將可利用已存用的市場途徑進軍美國，分析家預料日本製造的"CASHEWS"，"AL－MONDS"或IBM PEANUT模仿品逐步滲入美國。

SPECTRAVIDEO （美國的生產商，香港亦有廠）亦與十七家日本製造商達成新 MSX 家庭電腦標準，訂定軟件硬件應用標準，利用此標準的產品，主要是在日本市場推銷及力圖抗拒 IBM在日本市場的聲勢。

分析家預期家庭電腦市場將有更精細的劃分。新的家庭電腦產品售價會在二百至五百美元之間，可以玩遊戲控制家庭支銷及進行特定的工作，而低檔的私人電腦，如 PEANUT ，售價在五百美元至一千美元之間，是用作文字處理， SPREADSHEET 分析，數據庫管理，由於前一類電腦價格較平，肯定逐漸由日本所輸入的產品壟斷市場。

未來年間，家庭電腦的商戰會逐步升級，「蘋果」IBM，COMMODO－RE，COLECO，ATARI，及 TAXAS INSTRUMENT 這些家庭電腦生產商將會競爭白熱化，IBM 推出 PEANUT肯定令市場形勢改變，而根據消息說，IBM及蘋果會推出另一批價格更廉宜的產品，誰優誰劣，足影响整個大勢。■

配有6502處理器，不用RAM卡即可抄錄程式

WILDCARD 新版本

筆者在「電腦時代」第二期中曾爲大家介紹WILDCARD的用途，此卡主要的作用是可以用來抄錄任何程式或 DUMP 任何 HI-RES 的畫面。（詳情請翻閱「抄錄程式的硬件工具— WILDCARD」一文）。今期筆者要爲大爲介紹的是第二代的WILDCARD，WILDCARD PLUS。這張新版本硬件較舊卡體積大很多，舊卡有11.2公分× 6.8公分，新版本爲16公分×6.8公分。更重要的是WILDCARD PLUS 本身裝有一枚 6502 微處理器，一如APPLE Ⅱ 本身亦擁有一枚6502一樣。

卡上的 IC亦數目有增加，由九枚增至十五枚。

舊版本 WILDCARD 受到非議的地方是操作的速度，其抄錄程式所需的時間，較任何同類競爭的產品爲慢。而且在抄錄之前，必須先 BOOT 碟。新卡將可撤除這項步驟，使抄碟更直接，因爲 WILDCARD PLUS 本身所要的啓動程式，全燒錄在卡上的 ROM 內。因此，抄碟時間特別快捷，當然，這還得視被抄程式的長短而定。舊卡只能處理 48 K長度的程式，新卡則可抄錄長至128K的程式。

通常，WILDCARD PLUS 抄錄一個48K的程式只要8秒，64K程式則要12秒，而一個125 K的程式，只需25秒以下的時間來抄錄。

在使用 WILDCARD PLUS 時，可安插它在任何一個擴展槽，然後 BOOT 起被抄的程式磁碟，按動接連卡的接綫頭後，出現有七項選擇：從記憶中取用程式至工具程式。

「C」抄錄磁碟。 BOOT 起磁碟後，用一隻由WILDCARD PLUS 工具程式 I－NITIALISED過的磁碟置換，按C鍵即可。抄錄時間通常只需8至12秒不等。即使在沒有任何指示下，大家都可以進行。

「R」是從記憶中取用程式 這主要是爲遊戲者而設。

「A」是修改選擇。調校主機的記憶量並測查每個擴展槽中所插上的卡，及辨清有否使用上80 COLUMN卡。能夠設定 RAM 記憶的使用量，對於抄碟有極大幫助，因爲WILDCARD PLUS毋須將一個只有48K長度的程式當作128 K處理，這會節省許多時間。

「L」是 LOAD 另一張磁碟。

「B」是 BOOT 磁碟。當一張磁碟不能自動 BOOT 起是便用得着。

另外兩個選擇是「M」— MONITOR及「U」—工具程式。

除卻這些動能外，這張卡還能作甚麽呢？

工具程式「U」容許 SAVE 數個 BACKUP 在磁碟上。恢復 BASIC 程式，解釋機械語言列表， DUMP 任何高解像畫面。附表是新卡和舊卡的分別。

新卡擁有 6502微處理器意味着它可重新發展設計，相信不久將會有許多功能更動更廣的新版本WILDCARD 出現。不過直至現時爲止， WILDCARD 仍受着一定的限制。

這張卡可用於48K的 APPLE 或 64 K或 128 K的 APPLE Ⅱ。■

WILDCARD及WILDCARD PLUS 的特徵比較

特徵	WILDCARD	WILDCARD PLUS
IC數目	9枚	15枚包括 6502
呎吋	11.2×6.8公分	16×6.8公分
被抄程式長度	48K	128K
抄錄	磁碟上	燒錄在 ROM 內
抄錄軟件	25秒	8秒起（視程式長度）
是否需用RAM卡	要	不用
未來發展	沒有	有—在ROM中

128K RAM 卡的妙用
COPYWRITER

PETER

良久以來，抄錄沒有作保護軟件的最快方法是使用兩個磁碟機，需時只要25 秒。但兩部磁碟機的轉速必需一致，否則會出現誤差。再者，這抄錄方式對於電源供應器造成相當大的負荷，而原有的磁碟經多次的被磁頭讀取後，一定會造成不可補救的磨損。現在就要向讀者介紹一種更佳的抄錄方法。

OMEGA MICROWARE生產了一個叫 COPY W—RITER的抄錄程式，令到抄錄過程耳目一新。這新一代的抄錄程式需要到 RAMEX 或 SATURN的 128K 記憶擴展卡，但只需用一部磁碟機，如果同時使用 8 部磁碟機，則可以作高速度的連續抄錄，每一個拷貝只須 16 秒，如需要 VERIFY（查證），亦只需時22 秒，比 APPLE Q—UICK COPY還要快。倘如磁碟表面或磁碟機欠妥，這抄錄程式即時會告訴用者。由於數據被載入電腦後即能直接抄錄進新磁碟並停留在 RAM 裡，再作抄錄時亦不必再讀磁碟，所以在進行抄錄過程中，亦可同時進行載入（LOAD）或載出（ UNLOAD ）。但對於有保護的磁碟，這個 COR—YWRITER 就無能爲力了。

在使用 COPYWRITER 之前，記着先抄錄一個副本以防萬一。使用 128K RAM 可依照說明書使用。

整個抄錄程式的操作過程均需用者根據目錄指引去做，這可以說是一個好主意，因爲沒有人可以保證永不插錯磁碟，在這種操作過程中就可以發現插錯碟而避免損壞了主碟。由於這個 COPYWRITER 是逐條 TRACK 及 SEC—TOR 抄錄，所以如 CP／M，PASCAL等的軟件亦可照抄無疑，如果使用兩個磁碟機或以上，程式就可以直接由一部磁碟機抄往另一部而無需作任何停頓（即兩部磁碟機作同步操作）。

現時常見的誤差（ ERROR ）有三種：

（1）W誤差—磁軌未被正確寫入數據，通常是磁碟機轉速過快。

（2）STEP—當所有磁軌被寫入數據後，磁碟本身就會查檢磁頭的步進硬件是否妥當，如有不妥就會立即指示出來。

（3）WRITE PROTECT誤差—磁碟有寫入保護，不能寫入任何數據。

完成一次抄錄後，換入新碟，按 SPACE BAR 兩次即自動重復進行抄錄。

一點要提醒大家，128K RAM 卡在磁碟 EMUL—ATION MODE是可以當作第二或第三個磁碟機使用，所以如是重複抄錄一個程式，最好是將資訊儲在 128 K CA—RD 裡，這就相等於擁有一部高速磁碟機了，採用了這種方法，用者在作多次同樣的抄錄時，就不必忍受原有的磁碟被磨多次及等待兩部磁機在讀寫數據時的跳來跳去的頭痛了。

128 K RAM 卡可以做的工作比任何的擴展卡爲多，它可以作高速抄錄，DISK EMULATION，印字機緩冲記憶，記憶擴展等，且可以當作爲一部磁碟機的使用。O—MEGA 有一個預先載入（PRE—BOOT），可以將 SLOT 5 的 128 K RAM 卡轉變成爲僞 SLOT 6 ，如此一來，就可以當作一部磁碟機了。

COPYWRITER售價 99.95 美元，包括了一本詳盡的使用手冊。

讀者可去函 OMEGA MICROWARE 索取更多的資料。其地址爲：

OMEGA MICROWARE
222 SOUTH RIVERSIDE PLAZA
SUITE 2234
CHICAGO, IL 60606 U.S.A. ■

港府決定全面推行中學電腦課程

港府已計劃在未來三至四年內，撥款三千至四千萬元，購買三千多部微型電腦安裝在全港各官立資助及津貼中學，以全面推行電腦課程。

政府電腦資料處理發言人表示：現時在三十間中學試辦的電腦課程證實非常有用，故港府計劃在全港各中學及工業學校全面推行電腦課程。

在師資方面，本港各師範學院亦將相繼設立微型電腦，以幫助這些未來的老師對微型電腦有所認識，更使其在將來教學方面有所幫助。

發言人又稱：職業訓練局屬下的工業訓練學校最近亦準備添置一百部較大的微型電腦，使就讀於工程、學術科目及會計等科的學生就讀。

較早時，教育署發言人曾表示雖然此項電腦科試驗計劃大受學校及學生歡迎，並正考慮將此課程推廣至其他學校，但尚需考慮計劃實施首年的全面評估報告結果及經濟問題是否能解決等才作最後決定，最快亦須在本年九月才能實施此計劃。□

機械語言程式輸入方法

正

許多讀者來電服務部詢電腦時代創刊號內有些程式是不能 RUN。或在程式執行到某一處便被中斷。追根索源，我們了解到這些讀者只是不明白機械語言的輸入方式及如何 SAVE 落磁帶或碟。事實上，我們所刊登的所有程式，全部都是能即時執行的，斷不會有問題。當然，這並非表示電腦時代所登載的程式，每一個都是盡善盡美，事實上，這些程式，都有瑕疵存在，只是作者或我們因經驗及知識的限制而未能發覺。我們熱切希望讀者的指正賜教。假如各位讀者發覺這些程式能夠作進一步的修訂或改良，請不要吝嗇，立即寄來給我們，好讓我們能盡早刊出讓其他讀者分享。

言歸正傳，我們常發覺一些程式是被分成兩部份：一部分是 BASIC；而另外一部份則是機械碼（MACHINE CODE）語言程式。尤其是大多數遊戲程式，機械語言程式是形成整個遊戲構圖的主要部份。因此，要使程式得到順利運行，我們必須將程式的這兩部份正確地輸入電腦內。相信大部份讀者都懂得怎樣輸入 BASIC 程式部份，在這裡不再復述。至於機械碼程式部分，一些初入門的讀者却不太明白應該怎樣輸入電腦才對。其實輸入的方法十分簡單；但輸入機械碼程式之前，我們必須先進入電腦的監督系統（MONITOR）。假若讀者擁有的電腦是AP Ⅱ，則只須從鍵盤按入 CALL −151 指令就可從 APPLESOFT 進入監督系統；這時游標（CURSOR）的左方出現一個星號＊，這個星號就是監督系統的標示字，代表電腦已進入監督系統。這之後，我們就可以將正確的字組（BYTE）放進適當的位置裡。先按入地址，然後是冒號：；跟着便按入所需字組內容，再按下〔RETURN〕便可。例如我們想將 6 D這個字組放進9088這個地址內只需按入

＊9088：6D〔RETURN〕

假如需要將一系列的字組內容放進連續的地址內，我們無須將每個地址逐一按入，只須按入起始地址及冒號，然後將每個字組逐一按入，核對無誤後便可按下〔RETURN〕，這些字組便會自動被貯入連續的地址內。例如我們想將6D、00、D4、00…… 等字組分別放進 9088、9089、908A、908B……等地址內，只須按入

＊9088：6D 00 D4 00……

當一切字組正確輸入後，便可按下〔RETURN〕。有兩點讀者必須要留意的：第一，輸入的字組與字組之間必須以一空位分隔開，不能連在一起。第二，每次輸入的字組不能太多。當我們在電腦螢幕上輸入第七行字組時（大約是第82個字組），電腦會發出嘟、嘟聲，表示不能再繼續輸入，否則電腦會自動取消剛才的所有輸入字組。這時我們便要按下〔RETURN〕，所有字組便會被貯進各地址內。假如仍然有其他字組需要輸入，按照這方法繼續輸入便可。若要電腦顯示出你剛才所輸入的字組內容，只須按入起始地址然後按下〔RETURN〕。每按一次〔RETURN〕，電腦會將八個連續地址的字組內容顯示出來。假若我們想知道兩個地址之間的字組內容，亦可按入<起始地址>・<結束地址>然後按〔RETURN〕，電腦便會一次過將這兩個地址之間的字組內容全部顯示出來。例如我們想知道地址 9088 至94A0之間的內容，只須按入

＊9088・94A0 然後按

〔RETURN〕

當完成一切輸入後，我們當然希望將這些資料儲存起來，以便日後使用。讀者可選擇儲存在磁碟或磁帶上。假如讀者使用的是磁帶，方法是按入<起始地址>・<結束地址>W，然後按〔RETURN〕。例如我們想將地址 9088 至94A0之間的字組內容儲存在磁帶上，只須按下

＊9088・94A0W 然後按

〔RETURN〕

要從磁帶讀回這些機械碼資料，方法是按入<起始地址>・<結束地址>R，然後按〔RETURN〕。例如我們想將磁帶上的資料放在地址 9088 至 94A0之間，只須按下

＊9088・94A0R 然後按

〔RETURN〕

星號＊表示我們必須先進入監督系統，然後才可打入上述指令。有一點是值得一提的，就是我們可以將這些資料放在任何可供我們使用的地址內，只要起始地址與結束地址之間有足夠的空間儲存磁帶上的資料便可。但假若這個機械碼程式是 BASIC 程式的一部份，我們最好還是將它放在原來的地址上，以確保整個程式能順利操作。

假若讀者使用是磁碟，儲存的方法是按入 BSAVE <存檔名稱>，A＄<起始地址>，L＄<字組長度>然後再按下〔RETURN〕。例如我們要將地址 9088 至94A0 之間的字組內容用 SHAPE 這個存檔名稱來儲存在磁碟上，由於這裡共有 419（94A0 減 9088十六進制）個字組，所以要按照以下的方法輸入：

BSAVE SHAPE，A$9088，L$419 然後按〔RETURN〕

$ 符號以後的數字是十六進制的。假若讀者使用十進位數字，可刪去$符號。例如：

BSAVE SHAPE，A$9088，L1045 跟上面的指令句是完全一樣的。還有一點要提醒讀者的，就是在打

（下轉 P14）

門外漢買印字機

陸生

電腦時代創刊號所刊登之兩個應用程式「電腦時代月曆」和「COMPU—TEXT 文字處理」都是相當實用的程式，相信讀者都會同意。不過筆者覺得要將此兩個程式發揮至最大效用，似乎必須要具備一部印字機不可，否則亦只有徒嘆奈何！

在各類電腦週邊硬件之中，價錢最貴的算是磁碟機和印字機。驟眼看來，磁碟機的實用性遠較印字機爲高，印字機並非必需品。特別對於只有興趣玩遊戲程式的朋友。不過筆者以爲，要將你家中的微型電腦效用發揮至最高效用，印字機實在是必備的東西。

當然，各位電腦迷心裡都有個算盤，若然不是有強烈需要，不會輕易解囊。不過，無論如何，爲著令讀者對將來購買印字機時有所準備，筆者特別寫出這篇探討文章，同時附以與印字機有關連的幾項事物，希望令讀者能較全面及深入地了解如何選擇一部最合乎經濟原則而效用質素良好的印字機。

四類印字機

基本上，現時市面有四類印字機供你考慮。

（1）成形字體印字機，它類似打字機，有高爾夫球形或菊花碟的鍵頭，字體預先鑄在鍵頭上。這類成形字體印字機，在擊打色帶前，要經過一個較慢的機械程序去將"鍵頭"上的字體旋轉到正確的位置，然後才擊打。

這類印字機最大的優點就是它們的印字質素。此外，它可以有多種字體供選擇。

它的缺點是速度慢和不能同時利用多種字體編印。除非換上不同的印字頭。

（2）點陣式印字機，它是利用一些尖針刺在紙張上，而針與紙是隔着一條色帶的，這樣便可以將電腦的輸出，用這些有色的針孔表示出來。

點陣式印字機，可以任意印出不同字體而無需更換印字機頭。更可以印出放大字體做標題；或斜體字樣強調某句子。

點陣式印字機的主要缺點是印出來的字樣質素較字體式印字機爲差。

（3）另一種點陣熱感式印字機的印字原理與點陣撞擊式的相似。只不過它是利用熾熱的尖針直接烙在特別的紙張上，產生出字樣來。

以上所介紹的印字機都是價錢實惠和操作時很靜，但並不太適用於辦公室。

（4）用鐳射控制的印字機，會日漸受到微型電腦用家接受。它印字的原理是將墨水逐個字噴唧在紙上。

它具有極快的印字速度；多樣化印字方法和可以處理多種顏色的組合等優點。而其缺點是印字的清晰度不足和價錢高昂。

日本製造的印字機目前充斥市場。它們印字的質素都爲很多一般用家接納。它們印出來的字體，比例均勻和字體的清晰程度，令用家感到滿意。

不過有一點要注意的是日本製的印字機的字體，傾向於日文字形，故此當印羅馬字形時，顯得幼長些。

現時所購買的印字機，在使用方面都有一定的問題，這些問題都沒有在廣告或使用手冊提出，用家尤需注意，印字機是由軟件驅動的；換言之，在印字機內置程式和微處理器與微型電腦的程式和微處理器是有互通訊息。

若想保證你所購買的印字機能正常操作，最好是在購買之前，能親自看著它與電腦接駁後，正常地運作。

通常來說，印字機的使用手冊在編寫和內容安排上都很拙劣，亦沒有顧慮到普羅用家的需要。例如要去開動、停止或在句子下劃綫等，印字機都要得到從接駁著的電腦傳來了訊息，才會得操作，傳送這些訊息的方法，說起來應是很容易，實際上偏偏相反！

選擇適當的介面連接電腦和印字機亦是很重要的。若想印字機印出來的圖形與原圖一樣，就需要選擇一張能夠處理圖表的介面了。

另一種附加體對你也很重要的，若你的印字機是菊花碟形，它就是緩衝記憶體。有了它，電腦輸送往印字機的資料，可以儲存在它那裡，使電腦可以很快便空下來做其他的工作。

沒有緩衝記憶體，電腦便要待輸往印字機的資料完全印出來後，才可以去做另一樣工作。

印字機的速度由兩用的打字／印字機（每分鐘15—25個字母）至更快的菊花碟形印字機（每分鐘 80—100個字），最快的是點陣式印字機（每分鐘 200—400字）。

最後一點要考慮的是印字機的可印濶度和紙張的選擇。一般可印80個字濶度的，已足以應付日常需要。

很多時，你要用到是接連摺疊的紙張；可是有時你也會用單獨的紙張，例如提貨單之類。

現時有兩種紙張輸送法。一種是靠磨擦力；另一種是牽引式。用接連摺疊的紙張或標貼等，磨擦式的輸送法很易令到紙的位置移歪，故此應用牽引式輸送法的印字機爲妙。

當要購買印字機時，首先要做的是坐下來仔細想想：你將要用它來做甚麽；它的工作量將會是多少。很多情況下，若你沒有金錢的限制，購買一部印字機已足夠應付你的需要了。

不過印字機的速度與印字的質素，通常是成反比例的。故有時購買兩部印字機去處理不同的需要，可能比你的想像中的理想方法更便宜和更令你滿意呢！

印字速度的意義

目前印字速度快的點陣印字機和印字質素優的菊花碟形印字機，兩者之間的表現參差，正逐漸減少。這不是菊花

碟形印字機印字快了，而是點陣印字機印字的質素漸漸提高了。最通常與AP—PLE 配用的印字機，其中一隻便是EPSON系列——由RX 80 至FX80和 MX 100 F/I Ⅲ。但是，一些日本及台灣公司正提供低價的EPSON系列的抄製品。它們都聲明是全部與EPS—ON 配合的。

由於這些抄製的印字機，取得了用家的信任，故此給予原來的製造家一個威脅，而迫使他們割價競爭。

由EPSON FX80 以上的系列，基本的要求是全轉載濶度（132直行），高速度（一秒內超過 150個字母），良好的紙張處理，牽引及磨擦式兩用入紙裝設和更高密度的尖針點陣。

不過總有一天，點陣印字機會達到相當的清晰度，及兼具印圖案和利用軟件控制印字方式；或者印字的清晰標準降低了，令到點陣印字機的程度可以被接受的話，那麼點陣印字機便可成爲更有效率的硬本生產工具了。

菊花磚形印字機的製造家雖然用割價來競爭市場，但除非他們能在低價錢之餘，繼續維持印字的清晰度，否則他們的產品未許樂觀的。

印字機的選擇通常與整個電腦系統的耗費有密切關係，但很少用家肯樂意地付出較購買一部微型電腦爲多的價錢來購買一部印字機的啊！

保養

買回來後的印字機，必須要有好好保養。以下爲各項保養要點可供參考：

許多電腦用家過於信任硬件，而忽略了保養的重要性。他們認爲當硬件出現問題時，他們立刻可以得到維修，而不理會電腦系統停止所引致的損失。

這種想法可說是對的，但要視乎在甚麼條件下。事實上問題是很複雜的。如果一間機構的電腦設備是來自多個供應商的，那麼當硬 發生問題時，便有可能需要多過一個的供應商的去解決問題，因爲可能是由於一件硬件的故障而令到別的硬件也出了毛病。

印字機製造商通常只爲大的用戶提供服務，而且只得有限的工作間及維修分店，它們也許會爲你修理壞的機件，但可能需要時三十天。

在電腦系統內，印字機是最麻煩的。大多數的製造商只給與它三千小時保用期，即約等於每日連續使用八小時。在實際使用上，印字機的壽命當然可以更長，但一些製造 商正續漸收緊保用期的時間，有些更用印字機鍵鎚的敲擊次數來計算。

很多微型電腦用家都不依足印字機指南或微型電腦手册去操作，更不會作日常的保養檢查，只是在損壞後才給與檢查及修理。因而一部印字機的壽命亦大大的減短了。

紙張

使用印字機是不能沒有紙張的。現時市面究竟有多少種類的打字紙可供選擇呢？

目前市面最廣泛使用的是接連印字紙，有多種呎吋，有些是全白的，有些則間有橫綫（與五綫譜相似），後者在視覺上較易閱讀。

單張或是一式多份的紙料都有很多呎吋供選擇。例如一式四份的紙，適合商業性用途。有時爲了配合某些特設用途，需要使用預先印上格式項目的表格，使用時，只要印上姓名、地址、公司及有關資料等便可以，例如發票，提／出貨單等，我們稱這些爲預印表格。

我們慣常看見的電腦紙，兩邊都有一條可撕除的紙邊，上面有輪齒式的小孔（目的是方便印字機膠輪牽引）每條邊有半吋濶。以一張長 9 ½吋濶的電腦紙，實用面積只有8 吋半濶。

我們在選擇印字機時，必須要考慮到該印字機所能用到的紙張種類。例如能否使用單張普通紙，當然紙的濶度亦是重要的選擇。一般來說家用式的印字機，用 9 ½吋濶的紙便夠了，過濶只會浪費紙張。

緩衝記憶

印字機緩衝記憶，基本上是一個儲存由電腦出來的資料的容器。正常的情況下電腦輸出資料的速度是比印字機快的，往往，需要等待印字機完成所有印刷工作，才可以繼續使用電腦做其他工作，但使用緩衝就可以解決這個問題。你可以先將需要印出的資料輸送到印字機緩衝儲存，使印字機開始操作，於是電腦便可以同一時間作其他用途。而且當關掉電腦後，印字機仍可以獨自繼續操作。

一般緩衝器的功能有：

先讀先印

任意存取印字，可以隨意選擇程序然後合成印出，而不必經過人手修飾。

越過方式（BY PASS MODE）

緩衝器的記憶量一般由 8K BYTE 至128 K BYTE ，有些緩衝器有將資料壓縮的功能，可以將重印等。

越過方式，顧名思義當緩衝在這一個方式操作時，電腦就會直接將資料輸送到印字機，像以往沒有緩衝器時一樣。

印字帶

一般印字機所使用的印字帶有兩種選購：絲帶及炭帶。若論印字清晰程度，當然以炭帶爲最佳選擇。不過，由於炭帶只能使用一次，故此對家庭用戶來說是沒有必需。假如要求字體色素效果，當然是炭帶較佳。讀者購買之前，應考慮本身所需。小數最怕長計。目前市面最平的炭帶，一盒也要六、七十元。平均只能印十萬個字左右，便需要更新。市面上有些印字機是可以採用兩種帶任何一種，但部份則只提供一種。所以選購時必須要慎重考慮。■

□印字機專用詞滙

▲ 緩衝（BUFFER）

用以儲存由電腦輸送到印字機的資料，以待發出。這樣可以令電腦更快地空閒下來。緩衝愈大，就愈有效地將電腦空出來做其他工作。

▲ 菊花碟形字盤（DAISYWHEEL）

這是一些品質較佳的印字機採用的機械結構。是個高速轉動的碟環（塑膠或金屬）鑄有各種字模，當鍵鎚隔著色帶向正適當的字模敲擊時就會在紙上留下字樣，而圖形則可由小點慢慢的組成。

▲ 點陣式（DOT MATRIX）

這是一種印字方法，是一排的針刺向色帶而在紙上留下字樣，當針頭逐步的向橫移動時，就會有更多的字印出來，若針的數目越多越密，印出來的字就越美觀，DOT MATRIX 漸多用於繪圖方面。

▲ 印字速度（CPS）

在一秒鐘內印字的數目。這個數目與印字機本身的速度沒有絕對關係，因爲印字機速度通常是以最多的數字計算。

▲ 印字矩陣（PRINT MATRIX）

在點陣式印字機（或是類似的印字機）組成字的橫直點的數目，在同一部印字機印字及印圖形這個數目是不同的。

上紙方法（PAPER FEED）

印字機上紙有三種方法：

- 阻力上紙法（FRICTION FEE—D）：紙是由捲筒夾著，十分類似時下常用的打字機。
- 針上紙法（PIN FEED）：在捲筒的兩邊末端有些小針用以穿著紙邊的小孔拖動紙張。
- 拖拉機上紙法（TRACTOR FEE—D）：類似 PIN FEED 而是在捲筒上有細少的拖拉器（TRACT—OR），這些是可以向橫移動，以適合各種不同濶度的紙。

▲ 介面（INTERFACES）

這是一種電子物體，以便電腦與印字機可以互通訊息，但有很多種不同類形，當選擇印字機時要了解那個介面是否令你滿意。

▲ 下行字體（DESCENDERS）

留意字母如〝p，q，g，j〞等，一些印字機是不能將下半部印出來的，這都因爲印字機的效能低劣。

▲ 雙向（BIDIRECTIONAL）

部份印字機爲了增加印字速度，而不返回左邊的起點才再開始印，相反在印下一行時，是從右邊向左邊印。很多時因要印較爲濃密的字體，而只能單方印（從左到右）。

▲ 捲筒（PLATEN）

是一種滾筒用以帶動紙張，與打字機類似。

▲ 字體（FONT）

印字的各種不同形態，如斜體字（ITALICS），特大字（PICA），特小字（ELITE）。有些印字機是可以印幾種不同的字體的，有時可能要轉換菊花形碟。

▲ 打字／印字機

（TYPEWRITER／PRINTER）

有些電動打字機是可以接連到電腦去，因而可以得到印字機的效果，但是比較慢。

▲ 驅動軟體

（DRIVER SOFTWARE）

不需要任何新頴的介面，只是在遊戲埠（GAME PORT）控制印字機。有時用來印製圖案。

▲ 畫面繪製（SCREEN DUMP）

是將螢幕的一切，文字或圖案，全搬到印字機去。

▲ 字體質素

（LETTER QUALITY）

通常在紙上最好質素的文字，是菊花形碟，但目前漸漸有由點陣式印字機取代的趨勢。

▲ 熱感印字機

（THERMAL PRINTER）

是用一些發熱的小點在特製的紙上造出文字。類似點陣式的方法，但是並沒有色帶及移動針。這些印字機的價錢較便宜，但紙則較貴。而文字質素就不很理想。

▲ 噴墨（INK JEK）

將小點的墨水噴在紙上而造成文字，亦是類似點陣式印字機，但是沒有色帶及推動針。

▲ 資料傳送速率（BAUD RATE）

是資料經過一系列的介面，由電腦送到印字機的所需時間的量度單位。但要肯定印字機與電腦是配合的才可以測量。■

機械語言程式輸入方法（上接 P11）

入這句指令前必須先跳出監督系統，因爲在監督系統內是不會接納這樣的指令句的。跳出監督系統的方法是按〔R—ESET〕（有些 MODEL 要同時按〔CTRL〕及〔RESET〕）。

要讀回磁碟上的機械碼資料，只須按入 BLOAD <存檔名稱>，然後按〔RETURN〕。例如：

BLOAD SHAPE 然後按〔R—ETURN〕這樣，電腦就會自動將存檔 SHAPE內的資料放在它們原來的地址。假若我們要將它放在其它地址內，可按入 BLOAD <存檔名稱>，A$<起始地址>，然後再按〔RETURN〕。例如：

BLOAD SHAPE，A$8000

就是將 SHAPE 內的資料放在8000（十六進位）以後的地址內。讀者若使用十進位，可刪去$符號。

經過這番詳細介紹，相信大部分的讀者對機械碼語言程式的輸入、存儲及讀取方法都會有更深入的認識，希望各位能多加練習，日後再與各位討論機械碼語言程式一些較深入的問題。■

PARAMETER for LOCKSMITH 4.1

```
STAR BLASTER(PDS)
 00
S 07-20.5 BY 1.5...........72=00 73=00 77=00 78=00
                           79=12 7C=00 40=20 19=00
                           44=DF 45=AD 46=DE
```

COMPUTING AGE USES AN EASY-TO-READ FORMAT FOR ITS LOCK-SMITH PARAMETERS LIST. THE EXAMPLE ABOVE EXPLAINS HOW A PROGRAM SHOULD BE COPIED.

FOR THE GAME "STAR BLASTER," THESE STEPS SHOULD BE FOLLOWED:

1) COPY TRACK 00.
2) ALTER EACH PARAMETER LISTED TO THE NEW VALUE: CHANGE 72 TO 00, 73 TO 00, 77 TO 00, AND SO ON.
3) SET THE INCREMENT TO 1.5.
4) USE A SYNCHRONIZED COPY ON TRACKS 07-20.5.

SYMBOLS USED

S USE SYNCHRONIZED COPY.
...PARAMETERS TO BE CHANGED BEFORE COPYING THE TRACKS INDICATED.
BY INDICATES THE TRACK INCREMENT (USE 1 IF NOT OTHERWISE INDICATED).

WHEN FOLLOWING THE INSTRUCTIONS FOR A PROGRAM, COMPLETE EACH STEP IN THE ORDER LISTED. BE SURE TO CHANGE EVERY PARAMETER BEFORE COPYING THE TRACK(S) LISTED TO THE LEFT OF THE PERIODS. ONCE A PARAMETER HAS BEEN CHANGED, LEAVE IT AT THE NEW VALUE UNLESS THERE ARE LATER INSTRUCTIONS TO DO OTHERWISE.

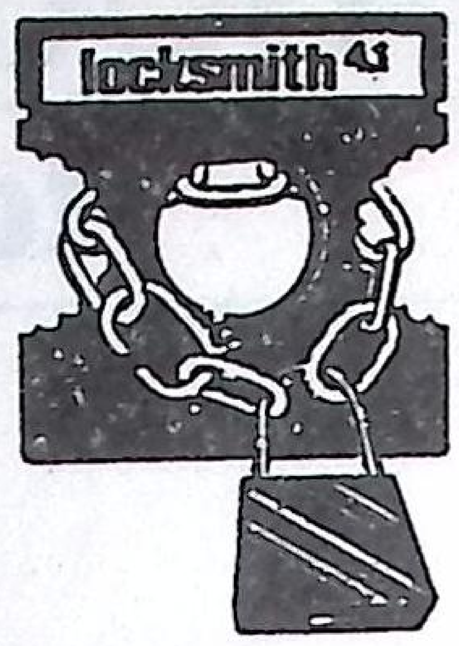
FOR ANY REMAINING QUESTIONS, CONSULT THE USER'S MANUAL FOR LOCK-SMITH.

IF A PARTICULAR PROGRAM IS NOT LISTED, TRY TRACKS 0-22 NORMAL. PROGRAMS THAT ONLY REQUIRE THIS ARE NOT LISTED DUE TO SPACE CONSTRAINTS.

THE ABBREVIATED COMPANY NAME CAN BE FOUND IN PARENTHESES TO THE RIGHE NAME OF THE PROGRAM. REFER TO THE TABLE OF ABBREVIATIONS FOR THE COMPLETE NAME OF THE COMPANY ■

AC--APPLE COMPUTER	IN--INSOFT	SOL-SIERRA ON-LINE
AG--AVANTE GARDE	MIS-MICROSOFT	SPC-SOFTWARE PUBLISHING CORP.
ART-ARTSCI	ML--MICRO LAB	SRS-SIRUS SOFTWARE
BC-BUDGECO	MU--MUSE	SS--SENTIENT SOFTWARE
BS-BRODERBUND SOFTWARE	PBS-PERSONAL BUSINESS SYSTEMS	STP-SOFTAPE
CC--CAVALIER COMPUTER	PDS-PICADILLY SOFTWARE	STS-STRATEGIC SIMULATIONS
CTS-CONTINENTIAL SOFTWARE	QS--QUALITY SOFTWARE	SVS-SILICON VALLEY SOFTWARE
DM--DATA MOST	RO--ROCKROY	SW--STONEWARE
HN--HAYDEN	SEN-SENSIBLE SOFTWARE	SY--SYNERGISTIC SOFTWARE
IC--INFOCOM	SL--SUB LOGIC	UNK-UNKNOWN
IDSI-IDSI	SIR-SIR-TECH	USA-USA
		VCP-VISICORP

LOCKSMITH 4.1

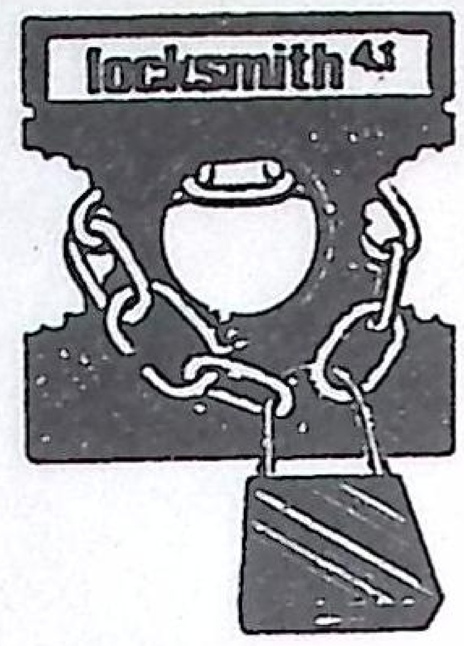



```
A2-FS1 (FLIGHT SIMULATOR)--(SL)
1) 00-21 BY 1.5
   07-08 BY 1
   9.5
2) 00
   1.5-21 BY 1.5....44:DB 45:AB  46:BF
40:20 4E:00 54:12
   07-08 BY 1
   9.5
A2-PB1 (NIGHT MISSION) -- (SL)
00
01-15.........44=DB 45=AB 46=BF 40=20
4E=00 54=12
WRITE PROTECT BEFORE RUNNING.
ALIEN RAIN (BS)
S 00-0E BY 1
APPLE LOGO (AC)
00-22
01............4C=1B 57=00 E9=02 34=FF
50=00 51=00 52=00 53=00
USES NIBBLE COUNT.
APPLE PFM (UNK)
00-22.........10=04 16=40 46=96 51=00
53=0B 54=12 81=CF 82=F3 83=FC
APPLE III BUSINESS GRAPHICS (AC)
S 00-22 BY 1..18=50 19=00 40=04 46=96
75=00 76=00 77=00 78=00 79=12
APPLE-WRITER II (AC)
00-22.........46=96 54=12
APPLE-WRITER III (AC)
S 00-22 BY 1
AUTOBAHN (SRS)
1) 00
   S 04-06 BY 1.....74=00
   S 09.5-0C.5 BY 1
2) 00
   04-06......74=00
09.5-0C.5
BAG OF TRICKS (QS)
00
01-14.........40=10 44=D6 53=00
BASIC MAILER (ART)
00-22.........4F=0B
BEER RUN (SRS)
1) S 00.......18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0D.5 BY 1
2) 00
   01.5-0D.5 BY 1...72=00 73=00 77=00
78=00 79=12 7C=00 40=20 19=00 44=DD
45=AD 46=DA
BORG (SRS)
1) S 00.......18=20 19=00 40=20 44=DD
45=AD 46=DA 72=00 73=00 77=00 78=00
79=12 7C=00
   S 01.5-0B.5 BY 1
   S 0D-20 BY 1
2) 00.........18=20 19=00 40=20 4D=00
4E=00 4E=00 52=00 53=00 54=12 57=00
72=00 73=00 77=00 78=00 79=12 7C=00
44=DD 45=AD 46=DA
   S 01.5-0C.5 BY 1
   S 0D-20 BY 1
BP I BUSINESS ACCOUNTING (AC)
1) 00-22......81=AD 82=FB 83=E6 84=FF
40=08 16=08 41=FF 19=00 58=0B 59=FF
2) 00-22......19=00 21=02 58=19 59=06
5A=1A 5B=FF BD=44 BE=E6 BF=45 C0=FF
C1=40 C2=01 C4=44 C5=D5 C6=45 C7=AA
C8=40 C9=04 CA=00
BRAIN SURGEON (UNK)
1) 00-22
   1B.........4C=1B 57=00 E9=02 D2=00
2) 00-22
   04.........4C=1B 57=00 E9=02 D2=00
BUDGE'S SPACE ALBUM (CP)
00-0B
BUG ATTACK (CC)
00-13 (0E-13 ERRORS MAY OCCUR)
1E............4C=1B 57=00 E9=02
USES NIBBLE COUNT.
CANNONBALL BLITZ (SOL)
00-22.........46=96 54=12 53=00
03-0F.........4C=1B 57=00 E9=02
USES NIBBLE COUNT.
CASTLE WOLFENSTEIN (MU)
1) S 00-22 BY 1.....46=B5 79=12
2) S 00-22 BY 1
COMPUTER AMBUSH (STS)
00
01-22.........4F=0B
COMPUTER CONFLICT (STS)
00
01-22.........4F=0B
COMPUTER NAPOLEONICS (STS)
00
01-22.........4F=0B
CONGLOMERATES COLLIDE (RO)
00-22
1B............36=01
CONGO (SS)
00-22.........46=96 4D=00 4E=00 21=02
26=06 51=00
CONTEXT CONNECTION (CON)
00-22.........19=01 21=02 58=19 59=06
5A=1A 5B=FF BD=44 BE=EB BF=45 C0=FD
C1=40 C2=01 C4=44 C5=D5 C6=45 C7=AA
C8=40 C9=04 CA=00
CRANSTON MANOR (HI-RES ADVENTURE #3)--
(SOL)
1) 00-22
   18.........4C=1B 57=00 E9=02
USES NIBBLE COUNT.
2) 00-22
```

LOCKSMITH 4.1

```
   18.........53=00 44=D5 45=FE 4C=1B
57=00
USES NIBBLE COUNT.
CYBERSTRIKE (SRS)
1) 00
   S 03-0B BY 1
   S 11-1C BY 1
2) 00
S 04-0B BY 1.....46=F5 79=12
S 11-1C BY 1.....46=B5
3) 00
   S 04-0B BY 1.....46=F5 79=12
   11-1C......46=B5
CYBORG (SS)
00-22.........47=FF 48=F8 4D=00 4E=00
51=00 40=04
DATA REPORTER (SY)
00-22.........4D=00 46=96 54=12
DB MASTER & UTILITIES (SW)
1) 00-05
   06.5-22.5 BY 1
2) 00-05
   06.5-22.5 BY 1
   WRITE PROTECT BEFORE RUNNING.
3) FOR VERSION 3.2:
   00
   S 01-05 BY 1
   06.5-21.5 BY 1
   22.5.......4D=00 46=96 54=12
DEAD LINE (IC)
00-22.........46=96 40=14
DESK TOP PLAN II (VCP)
00-22.........19=01 21=02 58=19 59=06
5A=1A 5B=FF BD=44 BE=EB BF=45 C0=FD
C1=40 C2=01 C4=44 C5=D5 C6=45 C7=AA
C8=40 C9=04 CA=00
DISK LIBRARY (UNK)
00-22.........40=09 53=00 16=77 46=96
47=AA 48=AA 4B=AA 54=12 21=02
DISK RECOVERY (SEN)
1) 00
   S 02-16 BY 1
2) 00
   S 02-04 BY 1
   S 0A-0B BY 1
DRAGON GAMES (UNK)
00-22
04............4C=1B 57=00 E9=02 D2=00
ELECTRIC DUET (IN)
00-22.........40=08 16=08 41=FF 19=00
81=DE 82=AA 58=0B 59=FF
ESCAPE FROM ARCTURUS (SY)
1) 00-22......4D=00
 00-22
2) S 00-22 BY 1.....4D=00
ESCAPE FROM RUNGISTAN (SRS)
S 00-21 BY 1..36=01
USES NIBBLE COUNT.
EXECUTIVE SECRETARY (PBS)
1) 01.5-21.5 BY 1
 00-22 BY 1
2) 00-22......46=96 54=12
EXPEDITER (SOL)
1) 00-22
 03 & 1F....4C=1B 57=00 E9=02
  USES NIBBLE COUNT.
2) 00-22
 03 & 1F....4C=1B 57=00 E9=02 D2=00
FINANCIAL CONTROLLER (UNK)
S 00-22 BY 1
GALACTIC SAGA I (EMPIRE) -- (BS)
S 00-23 BY 1
GALACTIC SAGA II (TRADER) -- (BS)
00-23
GALACTIC SAGA IV (TAWALA'S LAST
REDOUBT) -- (BS)
00............18=50 19=00 40=20 46=96
4D=00 4E=00 52=00 53=00 54=12 57=00
01-22.........44=D5 45=AA 46=B5
GALAXY WARS (BS)
S 00-12 BY 2
GAMMA GOBLINS (SRS)
00............18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
S 01.5-0D.5 BY 1....72=00 73=00 77=00
78=00 79=12 7C=00 40=20 19=00 44=DD
45=AD 46=DA
GENETIC DRIFT (BS)
00............18=50 19=00 40=20 46=96
4D=00 4E=00 52=00 53=00 54=12 57=00
01-03 BY 1....44=BB 45=B5 46=BB
04.5-06 BY 1.5
07.5-0B.5 BY 1
0D............44=D4 45=D5 46=BB
0E.5-12.5 BY 1......44=AD 45=B5 46=DE
GOBBLER (SOL)
00-22.........4E=00
03............4C=1B D2=00 45=DB 4E=01
34=FF 54=12 52=00
GOLD RUSH (SS)
00-22........46=96 4D=00 4E=00 21=02
26=06 51=00
GOLDEN MOUNTAIN (BS)
00
S 01-0D BY 2
02-0E BY 2
GORGON (SRS)
```

LOCKSMITH 4.1




```
1) 00..........54=12
   S 01.5-0E.5 BY 1............54=09
2) 00.........18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0E.5 BY 1.......72=00 73=00
77=00 78=00 79=12 7C=00 40=20 19=00
44=DD 45=AD 46=DA
HADRON (SRS)
1) S 00.......18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0D.5 BY 1
2) 00.........18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0E.5 BY 1.......72=00 73=00
77=00 78=00 79=12 7C=00 40=20 19=00
44=DD 45=AD 46=DA
HAYDEN ALIBI (HN)
00-02
03-22.........51=00 52=00 53=00 54=12
19=00 18=50 57=00 44=D4 46=B5
1B............4C=1B E9=02
USES NIBBLE COUNT.
HAYDEN APPLESOFT COMPILER (HN)
S 00-22 BY 1..46=96 71=19 79=12
ERRORS ON 10-1E O.K.
VERY SENSITIVE TO DRIVE SPEED.
HI-RES SECRETS (AG)
00-22........46=96 54=12 34=FB
HIRES SOCCER (SOL)
S 00-22 BY 1
HYPER HEAD ON (DEMON DERBY) -- (BS)
S 00-12 BY 2
IMAGE PRINTER (SEN)
S 00-07 BY 1
S 09-22 BY 1
08............4C=1B 57=00 E9=02 D2=00
44=FE 45=AB 54=12 50=00 51=00 52=00
53=00
JAWBREAKER (SOL)
1) 00-22
   03.........4C=1B 57=00 E9=01
   USES NIBBLE COUNT.
2) 00-22
   03.........34=FF 44=DF 45=EF 46=F7
50=00 51=00 52=00 53=00 54=12
MAD VENTURE (ML)
S 00-23 BY 1
MAGIC SPELLER (ART)
00-22.........4F=0B
MAGIC WORD (ART)
00-22.........4F=0B
MASTER DIAGNOSTICS PLUS (UNK)
00-22
04.............4C=1B 57=00 E9=02 D2=00
MASTERTYPE (LNS)
1) 00-02
   03-1A......44=D4 54=12
   1C-22
2) 00-02
   03-1A......44=D4
   1C-22
MILLIKEN MATH (ML)
00-22.........4C=18 46=B5 54=12 50=00
51=00 52=00 53=00
MISSION ASTEROID (SOL)
S 00-22 BY 1
MOUSKATTACK (SOL)
00-22.........46=96 54=12 53=00
23............4C=1B 57=00 E9=02
USES NIBBLE COUNT.
MONTY PLAYS MONOPOLY (IC)
00-05.........1E=0B
MULTI DISK CATALOG III (SEN)
1) S 00-02 BY 1
   S 04-09 BY 1
2) S 00-22 BY 1
MYSTERY HOUSE (HI-RES ADVENTURE #1) --
(SOL)
1) S 00-22 BY 1
2) 00-10
   12-22
NIGHTMARE GALLERY (SY)
00-22........46=96 54=12 51=00 4D=00
4E=00
OLYMPIC DECATHLON (MIS)
1) 00-22
2) S 00-22 BY 1.....46=B5 A8=00 71=18
79=12
OO-TOPOS (SS)
1) 00-22.....32=88 01=06
2) 00-22......21=02
3) 00-22......4D=00 4E=00 21=02 2C=06
48=EE 49=FF
OPERATION APOCALYPSE (STS)
00-22.........25=19 65=00 6B=00
OUTPOST (SRS)
1) 00.........18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0D.5 BY 1.......72=00 73=00
77=00 78=00 79=12 7C=00 40=20 19=00
44=DD 45=AD 46=DA
2) 00.........18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-09.5 BY 1.......72=00 73=00
77=00 78=00 79=12 7C=00 40=20 19=00
44=DD 45=AD 46=DA
```

LOCKSMITH 4.1




```
PADDLE GRAPHICS (SOL)
00-22
23............36=01
USES NIBBLE COUNT.
PEGASUS II (SOL)
1) 00-22
   03.........4C=1B 57=00 E9=02
   USES NIBBLE COUNT.
2) 00-22......4E=00
   03.........4C=1B 57=00 E9=02
   USES NIBBLE COUNT.
PFS (SPC)
1) 01-13
   00.........40=08 41=FF 16=08 19=00
58=0B 59=FF 54=12 12=02 44=93 45=F3
46=FC 47=FF 81=93 82=F3 83=FC 84=FF (00
ERROR MAY OCCUR)
   WRITE-PROTECT DISK BEFORE RUNNING.
2) 00-22......10=04 16=40 46=96 51=00
53=0B 54=12 81=CF 82=F3 83=FC
PHOTAR (STP)
S 00-22 BY 1
POOL 1.5 (IDSI)
1) S 00-15 BY 1
   S 1E-21 BY 1
2) S 00-15 BY 1.....46=B5 79=12
   S 1E-21 BY 1
PRESIDENT ELECT (STS)
1) 00-22......25=19 65=00 6B=00
2) 00-22......25=19 6B=00
PUCKMAN (UNK)
00............54=12
01-0D.........54=09
PULSAR II (SRS)
S 00
S 1C.5-1D.5 BY 1
S 02-0C BY 1..44=DD
S 13-19 BY 1
S 1A.5-1B.5 BY 1
QUICK LOADER (SEN)
00
S 02-11 BY 1
RASTER BLASTER (BC)
1) 00.........44=AD 45=DE
   S 05-11 BY 4
   S 06-12 BY 4
   S 07.5-0F.5 BY 4
   S 01.5-03.5 BY 2
2) 00.........46=96 54=12
   S 05-11 BY 4.....44=AD 45=DE 46=00
72=00 73=00 75=00 78=00 79=12
   S 06-12 BY 4
   S 07.5-0F.5 BY 4
   S 01.5-03.5 BY 2
RETROBALL (SOL)
00
04-06
09-0C
0E-10
12-14
17-1D
20-22.........4D=00 4E=00
RINGS OF SATYRN (SL)
S 00-02 BY 1
  03-22
S 05
S 09
SARGON II (HN)
1) 00-1A.....19=00 54=12 47=FF 4C=18
48=FF 50=00 51=00 52=00 53=00
2) 00-1A......19=00 54=12
SCREENWRITER II (SOL)
00-22.........4D=00
SHATTERED ALLIANCE (STS)
1) 00.........25=19 65=00
   01-22......4F=0B
2) 00-22.........25=19
3) 00.........4C=18 47=FF 53=0B 54=12
   01-22......44=D4 46=B5
4) 00-22......25=19
SNAKEBYTE (SRS)
00............18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
S 01.5-0A.5 BY 1....72=00 73=00 77=00
78=00 79=12 7C=00 40=20 19=00 44=DD
45=AD 46=DA
SNEAKERS (SRS)
00............18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
S 01.5-0C.5 BY 1....72=00 73=00 77=00
78=00 79=12 7C=00 40=20 19=00 44=DD
45=AD 46=DA
SNOGGLE (PUCKMAN) -- (BS)
1) 00-09
2) 00-0F
   S 10.5-11.5 BY 1
3) S 00-09 BY 1
STAR BLASTER (PDS)
00
S 07-20.5 BY 1.5....72=00 73=00 77=00
78=00 79=12 7C=00 40=20 19=00 44=DF
45=AD 46=DE
STAR THIEF (CC)
1) 00-0E ERROR MAY OCCUR ON 0E.
22.......4C=1B  57=00  E9=02
   USES NIBBLE COUNT.
2) 00-13
   ERRORS MAY OCCUR ON 0E-13.
```

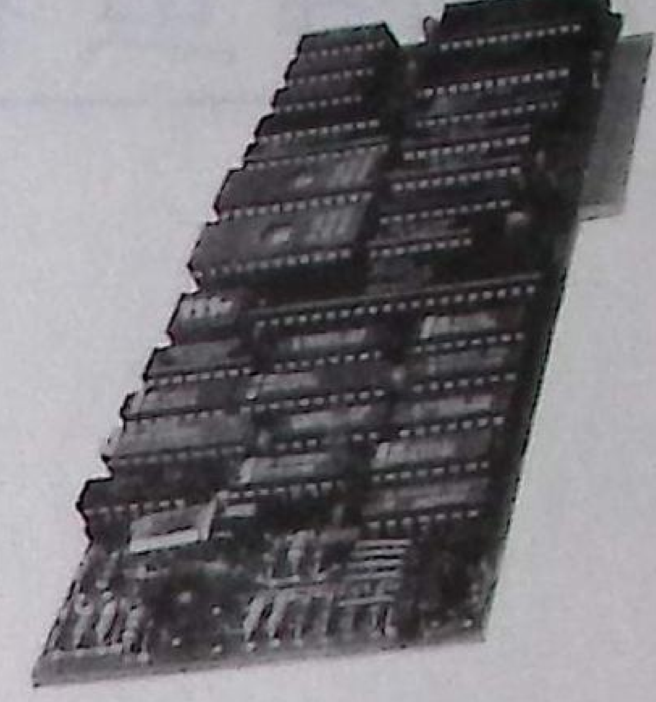
LOCKSMITH 4.1




```
   22.........4C=1B 57=00 E9=02
   USES NIBBLE COUNT.
THIEF (DM)
00-22.........83=FF 4F=0B 53=00
S 04-05 BY 1..38=02 1E=02 19=00 12=01
7C=00
THRESHOLD (SOL)
1) 00-22
   01-23 BY 22
   22.........4C=1B 57=00 E9=02
   USES NIBBLE COUNT.
2) 00-22
   01.........4C=1B 57=00 E9=02
   USES NIBBLE COUNT.
TIGERS IN THE SNOW (STS)
1) 00-22......25=19 65=00 6B=00
2) 00-22......25=19 6B=00
TIME ZONE (HI-RES ADVENTURE #5) --
(SOL)
1) S 00-04 BY 1
   05-22  (DISK SIDES 1B TO 6L,TRACKS
00-22.)
2) 00-22  USES EXTENDED RETRY. (DISK
SIDES 1B TO 6L, TRACKS 00-22.)
TWERPS (SRS)
1) 00.........18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0E.5 BY 1.......72=00 73=00
77=00 78=00 79=12 7C=00 40=20 19=00
44=DD 45=AD 46=DA
   1C.........4C=1B 57=00 E9=02 D2=00
2) 00.........18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0E.5 BY 1.......72=00 73=00
77=00 78=00 79=12 7C=00 44=DD 45=AD
46=DA
   S 1C
3) 00.........18=20 19=00 46=96 4D=00
4E=00 52=00 53=00 54=12 57=00 40=20
   S 01.5-0E.5 BY 1.......72=00 73=00
77=00 78=00 79=12 7C=00 40=20 19=00
44=DD 45=AD 46=DA
   1A.........4C=1B 57=00 E9=02
   USES NIBBLE COUNT.
U-BOAT COMMAND (SY)
1) 00-22......4E=00 51=00 52=00 40=02
1E=30 1B=19 1D=18 44=00 45=00 46=EB
47=AF
2) 00-22......4E=00 51=00 52=00 40=02
1E=30 1B=19 1D=18 44=00 45=00 46=EB
47=AF 48=FB 49=EB
VISICALC (VCP)
1) 00-22  IGNORE 01 ERROR.
2) 00-15  IGNORE 01 ERROR.
VISIDEX (VCP)
1) 00-22......40=04 16=08 41=FF 19=00
58=0B 59=FF 81=AA 82=EB 83=FD 21=02
2) 00-22......40=04 16=08 41=FF 19=00
58=0B 59=FF 81=AA 82=EB 83=FD 21=02
46=96 54=12
VISIFILE (VCP)
1) 00-22......19=01 21=02 58=19 59=06
5A=1A 5B=FF BD=44 BE=EB BF=45 C0=EC
C1=40 C2=01 C4=44 C5=D5 C6=45 C7=AA
C8=40 C9=04 CA=00
2) 00-22......19=00 21=02 58=19 59=06
5A=1A 5B=FF BD=44 BE=EB BF=45 C0=EC
C1=40 C2=01 C4=44 C5=D5 C6=45 C7=AA
C8=40 C9=04 CA=00
VISISCHEDULE (VCP)
00-22.........40=04 16=08 41=FF 19=00
58=0B 59=FF 81=AA 82=EB 83=EC 21=02
46=96 54=12
WIZARD AND THE PRINCESS (HI-RES
ADVENTURE #2) -- (SOL)
S 00-22 BY 1
WIZARDRY #1 (PROVING GROUND) -- (SIR)
1) 00-09
   0F-22
   S 0A-0E BY 1.....36=01
   USES NIBBLE COUNT.
   WRITE PROTECT BEFORE RUNNING.
2) 00.........36=01 21=02 46=96
   S 01-22 BY 1.....36=00
   USES NIBBLE COUNT.
   WRITE PROTECT BEFORE RUNNING.
WIZARDRY #2 (KNIGHT OF DIAMONDS) --
(SIR)
S 00-09 BY 1
S 0F-22 BY 1
S 0A-0E BY 1..36=01
USES NIBBLE COUNT.
WRITE PROTECT BEFORE RUNNING.
WORD HANDLER (SVS)
00...........46=96 54=12
11-22
01-0C.........44=FF 45=DF 46=DE
TYPE 8 ERRORS O.K.
WORD HANDLER II (SVS)
00...........46=96 54=12 53=00
11-22
01-0C.........44=FF 45=DF 46=DE
IF TYPE 8 ERROR HAPPENS RECOPY TRACK
UNTIL GOOD.
```

有十六種功能的 VIDEX ULTRATERM卡

螢幕顯示的魔術家



VIDEX推出一種新的週邊卡，稱爲 ULTRATERM。它除了有舊的 VIDEOTERM 的各種功能之外，還有其他特點，特點如下：

1、24一行×80　一列的顯示
2、24一行×96　一列的顯示
3、32一行×128 一列的顯示
4、24一行×132 一列的顯示
5、48一行×80　一列的顯示
6、由24一行×80一列操作的 VIDEOTERM EMULATION
7、正常／高解像度字符組
8、正常／反白顯示
9、HIGHLIGHT／LOWLIGHT字符
10、穩定，不閃爍的顯示，快速回卷軟件。
11、可以顯示全部96個 ASCII 字符，包括小階字符中眞正的下行小階字。
12、15字行繪圖，是標準字符組的其中一部份。
13、7字組圖案印刷，又是標準字符組的一部份。
14、7×9的點式字符印刷。
15、可以與任何週邊擴展槽一起運用，除了 SLET 0 之外
16、加上擴充了的記憶卡後，可以與 APPLE Ⅱe 一起應用。

最突出的是這種新卡的使用手册。每一樣東西都包括在內，如果看不明白的話，再重讀一次，你會發覺沒有一樣東西漏了。還有檢查出錯的特別說明。

視頻監視器的選擇

要儘量發揮新張卡的功能，最重要一點是選擇一部合適的監視器。筆者採用的是 NEC JB－1201 M綠色螢幕。如果你想得到一部合適的監視器，可以試試 AMDEK 300 A。AMDEK 300 A有較慢的綠磷，最適合這種新卡。問題是，在玩遊戲時，可能出現鬼影，這是因爲顯示轉變得快的緣故。這張卡，還有一個特別的160字符模式，不過，你需要調整大多數的監視器，減小調度才可以看到它有160個字符。還有，監視器要是15 MH2 或者更高。不過，筆者相信利用160字符作顯示的人必定很少。

這張卡安裝方法很容易，把卡插入 SLOT 3，其中一個 VIDEO CARD（MALE） 引入電腦，把監視器插入 FEMALE CARD。SOFT－SWI TCH在卡之中。

PREBOOTS

VIDEX正在製造一些適合例如 VISICALC，WOR－DSTAR與及 APPLE WRITER Ⅱ軟件的特別 PREBOOTS。

HIGH LIGHT ／LOW LIGHT

這可能是 ULTRATERM 最突出之處，此卡附同有一張示範磁碟，可以示範各種特徵。其中一樣是多種不同的字母、圖案，與及數據可以同時在螢光幕上顯示出來假如你不知道的話，會以爲是看雷射激光的表演 。解像度亦一樣如此。VISICALC示範令SPREADSHEET的可讀性更高，而它的額外文字行列又可以令你得到更多資料消息。

圖案模式

使用手册附錄中有詳細說明特別的圖形字符。GRAPHIC 的選擇似乎與舊的 VIDEOTERM無分別，事實上，這張新卡有很多舊的 VIDEOTERM 沒有的東西。

字符字體

有多種不同字符印字，甚至包括希伯萊及猶太文。

ULTRATERM 的指令

1. Control-A Uppercase/lowercase toggle
2. Control-K Generates the [character
3. Control-S Pause
4. Control-G Bell
5. Control-H Moves cursor back on space
6. Control-J Line feed
7. Control-K Output command for clear the display from cursor
8. Control-L Clears the whole screen
9. Control-M Carriage Return
10. Control-N Sets display attribute set
11. Control-O Alternate attributes command
12. Control-R Raw Mode-disables special commands
13. Control-U Sets normal Apple 40 column mode
14. Control-V has nine functions listed below:
    - 0 Apple 40 column
    - 1 80x24 video mode
    - 2 96-column by 24-line display mode
    - 3 160-column by 24-line mode
    - 4 80-column by 24-line interlaced characters
    - 5 sets the 80 by 32 mode with interlace
    - 6 sets the 80 by 48 more with interlace
    - 7 sets the 132 by 24 mode with interlace
    - 8 sets the 128 by 32 mode with interlace
15. Control-W Lead in character for video attribute

7	alternate char. set	inverse video	highlight
6	alternate char. set	inverse video	lowlight
5	alternate char. set	normal video	highlight
4	alternate char. set	normal video	lowlight
3	standard char. set	inverse video	highlight
2	standard char. set	inverse video	lowlight
1	standard char. set	normal video	highlight
0	standard char. set	normal video	lowlight

16. Control-Y Move the cursor to upper left-hand corner of display
17. Control-Z Lead in for control-character display command ■

旋轉螢幕畫面

各位讀者在看過本刊第一期的〝上下左右移動文字畫面〞一文後，對於螢幕捲動功能已有一定概念了吧。

這裡所刊列的程式再對旋轉功能作進一步的提升，各位不妨試試看。而且可加入至其它程式中使用，這裡所表列的機械語言可以隨意更改它們在記憶體內的位址，惟當 BASIC 程式 CALL 它們的位址亦要作相應的更改，不過應緊記在監督程式內的位址爲16進制，而在 BASIC 時所 CALL 的位址必須爲十進制，因此應予以準確對換，以下爲各子網路的開始位址。

	BASIC	MONITOR
螢幕下捲	CALL768	300G
螢幕左捲	CALL820	334G
螢幕向中心轉動	CALL857	359G
螢幕左右交替	CALL913	391G

以下的範例程式爲示範在 BASIC 程式內的使用方法，只須以 PRINT CHR＄(4)〝BLOAD SCROLLING SCREEN〞使可以。當單獨使用時只須 CALL 以上所列的位址便行了。

☆ 在示範程式中 SCROLLING SCREEN DEMO 中的鍵盤控制如下：

〔鍵〕	功能
←	左轉
→	隔行左右交替轉動
SPACE BAR	向下旋轉
C	向中心旋轉
ESC	停止程式執行

旋轉螢幕畫面程式

```
]CALL-151

*300L

0300-   A9 18      LDA   #$18
0302-   85 FE      STA   $FE
0304-   A2 17      LDX   #$17
0306-   8A         TXA
0307-   20 C1 FB   JSR   $FBC1
030A-   A5 28      LDA   $28
030C-   85 06      STA   $06
030E-   A5 29      LDA   $29
0310-   85 07      STA   $07
0312-   CA         DEX
0313-   8A         TXA
0314-   20 C1 FB   JSR   $FBC1
0317-   A0 27      LDY   #$27
0319-   B1 28      LDA   ($28),Y
031B-   91 06      STA   ($06),Y
031D-   88         DEY
031E-   10 F9      BPL   $0319
0320-   E0 00      CPX   #$00
0322-   F0 02      BEQ   $0326
0324-   10 E4      BPL   $030A
*L

0326-   A0 27      LDY   #$27
0328-   A9 A0      LDA   #$A0
032A-   91 28      STA   ($28),Y
032C-   88         DEY
032D-   10 FB      BPL   $032A
032F-   C6 FE      DEC   $FE
0331-   D0 D1      BNE   $0304
0333-   60         RTS
0334-   A9 28      LDA   #$28
0336-   85 FE      STA   $FE
0338-   A2 00      LDX   #$00
033A-   A0 00      LDY   #$00
033C-   8A         TXA
033D-   20 C1 FB   JSR   $FBC1
0340-   C8         INY
0341-   B1 28      LDA   ($28),Y
0343-   88         DEY
0344-   91 28      STA   ($28),Y
0346-   C8         INY
0347-   C0 27      CPY   #$27
*L

0349-   D0 F5      BNE   $0340
034B-   A9 A0      LDA   #$A0
034D-   91 28      STA   ($28),Y
034F-   E8         INX
0350-   E0 18      CPX   #$18
0352-   D0 E6      BNE   $033A
0354-   C6 FE      DEC   $FE
0356-   D0 E0      BNE   $0338
0358-   60         RTS
0359-   A0 14      LDY   #$14
035B-   84 FE      STY   $FE
035D-   A2 00      LDX   #$00
035F-   A0 12      LDY   #$12
0361-   8A         TXA
0362-   20 C1 FB   JSR   $FBC1
0365-   B1 28      LDA   ($28),Y
```

旋轉螢幕畫面程式（續）

```
0367-   C8          INY
0368-   91 28       STA   ($28),Y
036A-   88          DEY
036B-   F0 03       BEQ   $0370
*L

036D-   88          DEY
036E-   10 F5       BPL   $0365
0370-   A9 A0       LDA   #$A0
0372-   91 28       STA   ($28),Y
0374-   A0 15       LDY   #$15
0376-   B1 28       LDA   ($28),Y
0378-   88          DEY
0379-   91 28       STA   ($28),Y
037B-   C8          INY
037C-   C0 27       CPY   #$27
037E-   F0 03       BEQ   $0383
0380-   C8          INY
0381-   10 F3       BPL   $0376
0383-   A9 A0       LDA   #$A0
0385-   91 28       STA   ($28),Y
0387-   E8          INX
0388-   E0 18       CPX   #$18
038A-   D0 D3       BNE   $035F
038C-   C6 FE       DEC   $FE
038E-   D0 CD       BNE   $035D
*L

0390-   60          RTS
0391-   A2 28       LDX   #$28
0393-   86 FE       STX   $FE
0395-   A2 00       LDX   #$00
0397-   8A          TXA
0398-   20 C1 FB    JSR   $FBC1
039B-   A0 01       LDY   #$01
039D-   B1 28       LDA   ($28),Y
039F-   88          DEY
03A0-   91 28       STA   ($28),Y
03A2-   C8          INY
03A3-   C0 27       CPY   #$27
03A5-   F0 03       BEQ   $03AA
03A7-   C8          INY
03A8-   10 F3       BPL   $039D
03AA-   A9 A0       LDA   #$A0
03AC-   91 28       STA   ($28),Y
03AE-   E8          INX
03AF-   8A          TXA
03B0-   20 C1 FB    JSR   $FBC1
```
■

示範程式

```
]LIST

10  PRINT  CHR$ (13); CHR$ (4);"B
    LOAD LEAVING HOME"
20  HOME
30  FOR I = 1 TO 24
40  PRINT "COMPUTING AGE COMPUHOU
    SE CONTRIBUTION";
60  NEXT I
100  GET A$
110  IF A$ =  CHR$ (8) THEN  CALL
    820: GOTO 20
120  IF A$ =  CHR$ (21) THEN  CALL
    913: GOTO 20
130  IF A$ =  CHR$ (32) THEN  CALL
    768: GOTO 20
140  IF A$ =  CHR$ (67) THEN  CALL
    857: GOTO 20
150  IF A$ =  CHR$ (27) THEN  END

160  GOTO 100
```
■

DOS殺手

增加磁碟 8K BYTE 貯存量

□LUK

□ （一）引子

在未談這個程式之前，讓我們先來重溫 APPLE Ⅱ 的磁碟結構。在這 5 ½吋的磁碟的表面，它可以劃爲 35 個磁軌，最近圓周爲磁軌 0 （ TRACK 0 ），而最近中心的爲磁軌 35 （ TRACK 35 ），而每一個磁軌又可劃分爲16個扇區（ SECTOR ），每一個扇區所能儲存的資料爲 256 字節（ BYTE ），換句話說，一隻磁碟的總容量爲256×16×35＝140 K BYTES 。但實際的資料容量則只有31個磁軌，其原因就是 APPLE Ⅱ 把它的磁碟操作系統（ DOS ）放在前面的第 1 至第 2 個磁軌中，而它的 VTOC （ VOLUME TABLE OF CONTENT ）則放在第17個磁軌中，故此有 4 個磁軌是一早被 DOS 所佔用。APPLE Ⅱ把它的 DOS 放在第 1 至第 2 個磁軌中，是方便它的 BOOT UP 過程，但這卻平白地浪費了32個扇區殊不實際，以下的程式就是試圖把這個 DOS 抹去，慳回32個扇區共 8 K的儲存量。

要把這個DOS 抹去，我們把改變了的情形反映給 VTOC 知道，以便 VTOC 知道 1 及 2 磁軌爲可用的，但 DOS 抹去了之後，惟一帶來的缺點就是這張磁碟不可以再 BOOT 起，換句話說，我們一定要先用 DOS MASTER BOOT 起 DOS 3.3的操作環境，才可使用這張修改過的磁碟，但這個步驟只需做一次便可以。而其相對帶來的優點，就是我們現在擁有 528 個扇區的儲存量。

□ （二）程式簡述

這個程式是用 APPLESOFT BASIC 寫成，其工作原理，就是首先要把 IOB 表放在記憶區之地址 768—800 （ $300—$320 ）中。因爲IOB 表是DOS 去讀或寫扇區時所必需的，（詳情可參閱 APPLE 磁碟操作系統手冊）。當程式執行時，它會問那一個擴展插座（例如 6 ）及第幾個磁碟機（例如 1 ），這些資料就會被放在行號 160 中。跟着它會把磁碟的 VOLUME NUMBER變爲零（ 0 ），這樣便可與其他不同 VOLUME NUMBER 的磁碟相配合。跟着程式便把〝 255（$FF）〞這個數值寫在位於第17磁軌 0 扇區的 DIRECTORY 中的第 1 及 2 磁軌相對的 BIT MAP 上，從而令到 VTOC 以爲第 1 及第 2 磁軌是空的。（這是因爲在 APPLE Ⅱ 的磁碟操作系統中，每一條磁軌及扇區的佔用與否，皆由一個 BIT MAP 所控制，而這些 BIT MAP就是位於 DIRECTORY 中，讀者如欲更進一步了解的，可參閱〝 BENEATH THE APPLE DOS 〞一書）。

下一個步驟就是把第 1 及第 2 個磁軌全部重寫爲 0 ，程式中的行號 380 是把零的數值儲存，而在行號 400—470 是把這些〝零〞放出來寫在這兩條磁軌上。在每一次的讀或寫的操作後，程式都會跳往行號 520—590之偵錯子程式中，以便檢查程式的執行是否正確。

□ （三）偵錯程式

行號 520—590 行中之偵錯程式會監察着整個操作的進行。例如在地址值〝 791（$317）〞中發現其值爲〝 0 〞，則這個程式便會被觸發。假如在地址〝791〞中其數值爲〝16〞則表示這張磁碟是〝唯讀〞（READ ONLY），其他的錯誤會由DOS 自己去處理，只要改正了錯誤，再執行這個程式便可，當程式執行完畢時，它會再問你是否繼續，如答案爲〝Y〞，程式便會由頭做起，如〝N〞，則程序便會終止。□

```
1  REM ***************************
2  REM *    THE DOS KILLER       *
3  REM *    BY  THOMAS LUK       *
5  REM ***************************
10  TEXT : HOME : CLEAR :D$ =  CHR$
     (4)
20  REM  POKE IN IOB TABLE
30  RESTORE : FOR J = 768 TO 800:
      READ I: POKE J,I: NEXT J
40  DATA  169,3,160,10,32,217,3,9
     6,0,0,1,96,1,0,0,0,27,3,0,32
     ,0,0,1,0,0,96,1,0,1,239,216,
```

```
    0,0
50  PRINT "     DOS KILLER FOR 16
     SECTOR DISKS"
60  PRINT : PRINT
70  PRINT " THIS PROGRAM WILL REM
     OVE DOS FROM": PRINT "TRACK
     $1-$2 AND MAKE THEM AVAILABL
     E": PRINT "FOR DATA STORAGE"
     : PRINT
80  POKE 34,8: HOME
90  PRINT "ENTER DISK DRIVE SLOT?
     ";
100  GET SL$:SL =  VAL (SL$): IF
     SL < 1 OR SL > 7 THEN 100
110  PRINT SL$
120  PRINT : PRINT "ENTER DISK NU
     MBER";
130  GET SD$:SD =  VAL (SD$): IF
     SD <  > 1 AND SD <  > 2 THEN
     130
140  PRINT SD$
150  HOME : PRINT "     SLOT  ";SL
     : PRINT "    DRIVE  ";SD: POKE
     34,11: HOME
160  REM  POKE SLOT AND DRIVE
170  POKE 779,SL * 16: POKE 793,S
     L * 16: POKE 780,SD: POKE 79
     4,SD
180  REM  VOLUME OF 0
190  POKE 781,0: POKE 792,0
200  REM  CLEAR ERROR
210  POKE 791,0
220  REM  SET TO READ, CODE = 1
230  POKE 790,1
240  PRINT : PRINT "INSERT DISK I
     N DESIGNATED DRIVE": PRINT :
     PRINT "AND PRESS <SPACE> TO
     START, OR": PRINT : PRINT "
     <RETURN> TO EXIT": HTAB 20
250  GET A$: IF A$ <  >  CHR$ (32
     ) AND A$ <  >  CHR$ (13) THEN
     250
260  IF A$ =  CHR$ (13) THEN  TEXT
     : END
270  PRINT
280  HOME : PRINT "READING VTOC,
     17,0"
290  POKE 782,17: POKE 783,0: POKE
     786,0: POKE 787,32
300  CALL 768
310  GOSUB 520: REM  CHECK FOR ER
     ROR
320  REM    MARK VTOC LOCATIONS $3
     C-$43
330  REM   AS BEING EMPTY
340  FOR X = 60 TO 67 STEP 4: POKE
     8192 + X,255: POKE 8193 + X,
     255: NEXT X
350  HOME : PRINT "WRITING VTOC"
360  POKE 790,2: CALL 768: GOSUB
     520
370  HOME : PRINT "CLEARING TRACK
     S $1-$2"
380  FOR X = 8192 TO 8448: POKE X
     ,0: NEXT X: REM   MAKE A PAG
     E OF 0'S TO STORE WHERE DOS
     WAS
390  POKE 786,0: POKE 787,32: REM
      BUFFER
400  POKE 790,2: REM   SET TO WRI
     TE, CODE = 2
410  FOR X = 1 TO 2
420  FOR Y = 0 TO 15: REM   CHANG
     E TO 12 FOR DOS 3.2
430  POKE 782,X: REM    TRACK
440  POKE 783,Y: REM    SECTOR
450  CALL 768: REM    EXECUTE WRIT
     E
460  GOSUB 520: REM    CHECK FOR
     ERROR
470  NEXT Y: NEXT X
480  HOME : PRINT : PRINT "  ANOT
     HER?";
490  GET A$: IF A$ <  > "Y" AND A
     $ <  > "N" THEN 490
500  PRINT A$: IF A$ = "Y" THEN  HOME
     : GOTO 160
510  TEXT : END
520  IF  PEEK (791) = 0 THEN  RETURN
530 E =  PEEK (791)
540  IF E = 16 THEN  PRINT : PRINT
     "DISK IS WRITE PROTECTED"
550  IF E = 32 THEN  PRINT : PRINT
     "VOLUME MISMATCH"
560  IF E = 64 THEN  PRINT : PRINT
     "DRIVE ERROR"
570  IF E = 128 THEN  PRINT : PRINT
     "READ ERROR"
580  IF E <  > 16 AND E <  > 32 AND
     E <  > 64 AND E <  > 128 THEN
     POKE 791,0: RETURN
590  PRINT  CHR$ (7); CHR$ (7): TEXT
     : END
```

磁頭如何準確落在所需的SECTOR上

DOS的自動同步追踪器

閣下曾否對 APPLE 的 DOS 產生疑問……它如何使磁頭落在要讀或寫的磁軌中之準確位置呢？

要解釋這個問題，我們先要明白它能做到這一點是由於磁碟驅動器是採用步進式馬達及磁頭在某一磁軌中落下時它先經過一條特定的磁軌，說起來好像很輕而易舉，但我們不妨想一想，以300 RPM 的高速下 DOS 如何能在該磁軌中準確地落到所需的 SECTOR裡呢？答案是當磁碟被 INITIALIZE 時，DOS 會把一串數據及地址錄進每一磁軌中，因此當讀回它們時，便可以認出那一個 SECTOR 的位置，不過眞正的核心問題是：它如何知道這些標誌著位址的 BYTES 開始了呢？

我們先作一些基本討論：在磁碟上所錄進的二進位資料是一連串的1和0的訊號，表面上沒有告知一個 BYTE 的開始及結束，但考慮以下的 BIT串流。

10010100011101001110……

若我們從這 BIT 流的最左方開始，將它們分爲8 BIT 一組的 BYTE，將有如下的結果出現。

10010100＝$94
01110100＝$74
110……等

若我們在一個 BIT 後開始，數字値將變成

00101000＝$28
11101001＝$E9
10 ……等

很明顯，結果是不同的，因此並無顯著的標明這些 BIT 由那裡開始，但是問題是必須解決的，我們由兩個知道的事實著手。第一個是最左方的 BIT爲1的 BYTE 有效，用來讀入或寫出去磁碟上。第二個方法是令電子部份能找到一個 BYTE 的正確開始位址，這就是所謂自動同步的 BYTE，這種自動同步的 BYTE 是以 $FF 來作爲與其它 BYTE 的分別，它除了由8個 BITS 組成以外，它還額外多加入兩個 BIT，這兩個 BIT是零，因此自動同步BYTE 組的形式是 1110011 1111110011111111001111111110011 1111110011……等。

當磁碟被 INITIALIZED 時，在一條磁軌上的數處會被錄上如上形式的自動同步BYTES。亦因爲硬件部份會漠視在開始的0，故此 DOS 僅須讀入5個連續的自動同步 BYTE 便可以正常地讀取其它貯存了數據的 BYTE 了。

在本程式的功能領域裏，還包含了一個容許你任意調校磁頭位置的功能，本程式可容許你將磁頭放在自動同步 BYTE 組內任何位置。

本程式可以加入到其它大型的程式裡，例如以高解像度圖形輸出磁碟內的資料。控制磁頭位置的方式是經由遊戲棒及按下 PADDLE（0）來達成，而且程式本身亦可以很輕易加以改動以使用鍵盤控制，只需更改210，220及600，610 行便行了。

當程式執行後，重新按入遊戲棒的（0）會重新執行，按下 SPACE BAR 可終止本程式。□

DOS 自動同步追踪器程式

```
10  REM
15  TEXT : HOME
20  HTAB 7: VTAB 2: PRINT "*** AP
    PLE SELF-SYNC DEMO ***"
30  VTAB 4: HTAB 6: PRINT "(SELF-
    SYNC BYTE = 1111111100)"
40 B$ = ""
50  REM
60  FOR I = 1 TO 10
70 B$ = B$ + "1111111100"
80  NEXT
90 X = 1:B = 0
100  GOSUB 640
```

```
110  VTAB 13: HTAB 6: PRINT "ADJU
     AT PADDLE 0 TO POSITION"
120  HTAB 8: PRINT "THE HEAD INDI
     CATOR (^) "
130  PRINT : PRINT : PRINT : PRINT

140  HTAB 2: PRINT "THEN PRESS BU
     TTON(0) TO READ A BYTE."
145  FOR PAUSE = 1 TO 3000: NEXT

150  REM
160 H =  INT ( PDL (0) / 10) + 1
170  IF H > 20 THEN H = 20
180  VTAB 7: HTAB H:Q = H + 7
190  PRINT "      ^      "
200  REM
210  IF  PEEK ( - 16287) < 127 THEN
     160
220  IF  PEEK ( - 16287) < 127 THEN
     160
230  PRINT : CALL  - 958: REM
240  REM
250  VTAB 8
260  FOR I = 1 TO 5
270  HTAB 10
280 BYTE$ =  MID$ (B$,7 + H + 8 *
     I,8)
290  REM
300  REM
310  IF  LEFT$ (BYTE$,1) = "0" THEN
     H = H + 1:X = X + 1: GOSUB 7
     20: GOSUB 640: GOTO 280
320  REM
330  REM
340  VTAB 8 + I:SV = 9 + I:SH = 1
     3: PRINT : PRINT "BYTE $ ";I
      + 1;" IS ";
350  FOR K = 1 TO 8
360  VTAB SV: HTAB SH: PRINT  MID$
     (BYTE$,K,1);:SH = SH + 1:X =
     X + 1: CALL  - 198: GOSUB 64
     0
370  NEXT
380  REM
390  GOSUB 690
400  IF B = 1 THEN 440
410  IF  PEEK ( - 16287) < 127 THEN
     410
420  GOTO 450
430  REM
440  FOR K = 1 TO 500: NEXT
450  NEXT
460  REM
470  SPEED= 150
480  VTAB 16: HTAB 3: PRINT "SYNC
     HRONIZATION IS GUARANTEED AF
     TER"
490  VTAB 18: HTAB 10: PRINT "FIV
     E BYTES ARE READ."
500  FOR K = 1 TO 5000: NEXT
510  VTAB 16: HTAB 2: PRINT " AFT
     ER SYNCHRONIZATION IS ACHIEV
     ED    ": PRINT
520  PRINT "DOS LOOKS FOR  $D5 AA
      96 WHICH BEGINS": PRINT
530  PRINT "EACH ADDRESS FIELD. I
     T THEN READS": PRINT
540  PRINT "THE ADDRESS TO FIND O
     UT WHERE IT IS ON": PRINT
550  HTAB 15: PRINT "THE TRACK.";

560  SPEED= 255
570  REM
580  POKE 49168,0: REM
590  IF  PEEK ( - 16384) > 127 THEN
      POKE 49168,0: HOME : END
600  IF  PEEK ( - 16287) < 127 THEN
     590
610  IF  PEEK ( - 16287) < 127 THEN
     590
620  GOTO 10
630  REM
640  REM
650  HTAB 1: VTAB 6: PRINT  MID$
     (B$,X,40)
660  SPEED= 255
670  RETURN
680  REM
690  IF BYTE$ = "11111111" THEN  VTAB
     SV: HTAB 22: SPEED= 255: FLASH
     : PRINT "*** BINGO ***": NORMAL
     : CALL  - 958
700 B = 1
710  RETURN
720  INVERSE
730  REM
740  HTAB Q: VTAB 5: PRINT "V IGN
     ORE ZERO": NORMAL
750  FOR J = 1 TO 30: FOR K = 1 TO
     20: NEXT :Y =  PEEK ( - 1633
     6): NEXT
760  HTAB Q: VTAB 5: PRINT "
              "
770  RETURN
780  PRINT  PEEK ( - 16287): GOTO
     780
```

PASCAL與APPLESOFT的比較

邵炳均

在開始之時（大約三十年前），所有程式都是利用機器語言。程式編寫員要製作一系列的數據然後由電腦處理器閱讀而產生及採取一些有效行動。這種數據式的程式編寫法是非常困難繁複而沈悶，滙編語言（ASSEMBLY LANGU－AGE）基本上與此也相同，只不過一些數據可給予「名稱」及「簡寫」而已。

直到 FORTRAN（FORM－ULA TRANSLATOR方程式翻譯器的縮寫），已是較複雜的程式，它的主要作用是可作數學計算指示，看起來較似方程式，而不像是一連串機器指示。這是第一種高階程式語言，而其間的發展已經歷了不少苦心研究及化費了大量的時間。

其後，新概念不斷湧現，已設計不同類型的新語言，各有不同的特色與功能。BASIC 語言模仿了 FORTRAN，特別是爲了小型系統及初學程式編寫員而設計。ALGOL語言，則爲電腦科學家設計，而達到更高層次的抽象研究。FORTRAN 比機器語言更爲抽象，因爲每一項的操作過程都沒有明顯的指示：

F＝Q1*Q2／（4*P1*EO*（R**2））

就等如一連串的機械指示。ALGOL比FORTRAN更抽象，因爲機械上操作更爲隱蔽：特別的控制字眼如 WHILE 取代高深莫測的 GOTO，而數據並不一定要是數字，但可以思考 TRUE FALSE價値，字符串 ，及其他簡單類型的複雜組合。ALGOL，則有更多的竅妙及騙人把戲玩意及危險的特徵。

PASCAL語言是由 NIKL－AUS WIRTH 所創作，是特爲學生而設計的先進而安全的語言，它保留很多 ALGOL 抽象部分，但限制於精要微細部份，使程式容易閱讀。

PASCAL 在聖地牙哥加利福尼亞洲大學的微型電腦系試用。PASCAL語言加上 UCSD 操作系統已變成愈來愈普及，而APPL－E PASCAL 是該系統一個版本。

由於 APPLE 本身已有 B－ASIC 語言，爲甚麽還是買額外 16K RAM卡及 APPLE PA－SCAL 磁碟及使用手冊？難道 PASCAL比 APPLESOFT BASIC 更佳，更値多化一筆錢？

沒有一種程式不能以上述任何一種語言寫出來，分野只在寫程式及操作時的速度快慢而已，而其中又要視乎程式編寫員的風格而非單以兩種語言技術性分別而作評價。所以對於解答上述疑問時，我們首先討論〝PASCAL〞與〝BAS－IC〞不同之處。

APPLESOFT 是由APP－LE DOS 操作系統控制運作。UCSD 操系就不同，兩者之差別，最基本是每一種系統的磁碟都不能互用。APPLE DOS 隨時聆聽接受命令（CATALOG，LAOD等等）UCSD 好像有好多間房的屋（看圖1），你必須要在適當的房間內才能發出命令。

BOOT 起了 PASCAL 磁碟，等如你到了前門，處於發號司令的位置，你可以編輯或操作已寫成的程式，或進入另一個房間。螢幕上頂部出現提示字句列出你的選擇。你只須打個字便代表你的選擇。

緊記 PASCAL程式一定要先翻譯，即是翻譯成爲可以執行的模式。並要在運作前一齊做妥，而並不是像 BASIC 程式，可以在操作時停下來又譯一句。打〝C〞代表將源程式（文字檔案）轉爲執行程式（代碼檔案）打〝X〞即執行代碼檔案，打〝R〞即是先翻譯後執行。這是三個最重要的命令行動。

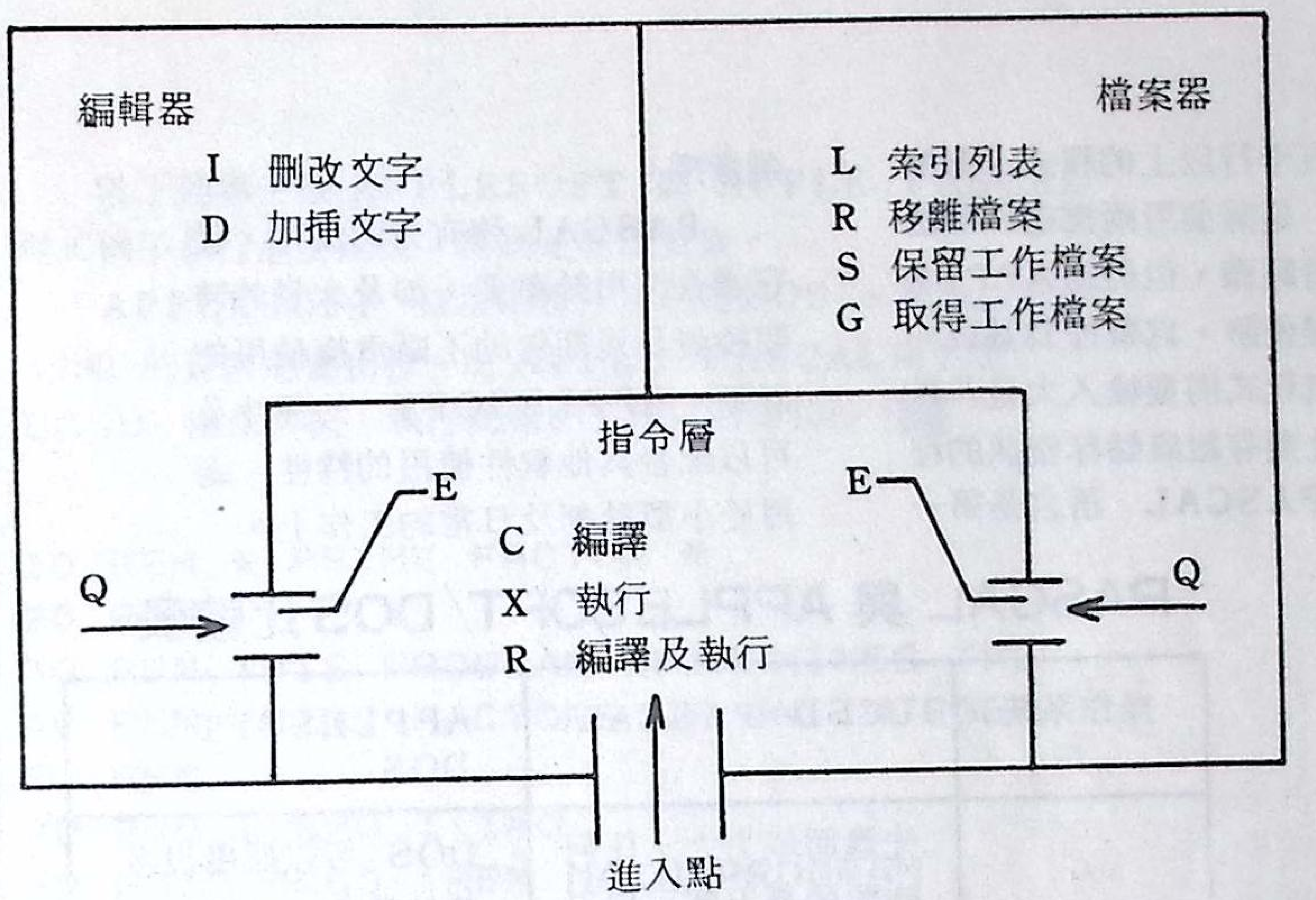
打〝E〞就將你轉到新的房間：編輯器（EDITOR）。這裡可打出 PASCAL 語言甚至打任何東西，由於 PASCAL 語言，是要經過翻譯才能應用，所以你可隨意打FOR－TRAN 程式，編譯語言程式，甚至是愛情信，編輯室都可隨你的愛好。因爲只有這房間才能進行編輯工作，放此編輯室比 APPLESOFT 先進得多，事實上，它也是全能的文字處理器。

命令級的〝F〞代表檔案儲存（FILER）它的多種功能與A－PPLE DOS 所提供一樣。你可將檔案儲入磁碟（按〝L〞掣）或取消（按〝R〞掣），查閱及保留在編輯器工作檔案（分別按〝G〞及〝S〞掣）這個工作檔案在磁碟上列爲特別檔案，〝SYSTEM. WRK.TEXT.〞編輯器及翻譯器可以自動利用這檔案。

無論你利用 UCSD PAS－CAL 語言到那個階段，螢幕上都出現指示字句，使你隨時清楚可作出那些命令。

這已顯示出 UCSD PAS－CAL 與 APPLESOFT／A－PPLE DOS 操作形式上不同。APPLESOFT／DOS 使你得到單一指示，就可以做任何事，只要你懂得如何打字輸入資料。但用 USCD PASCAL，你要化多些時間出出入入幾個房間，但你的選擇是非常明顯（螢幕出現指示），而效能更大（編輯器及編譯器功能

PASCAL與APPLESOFT的比較



圖（1） UCSD操作系統的〝樓面〞圖

比BASIC功能更廣，而檔案儲存器亦比DOS優勝超越）。

UCSD 操作系統與PASC—AL是獨立，但不如DOS接連APPLESOFT一様。PASCAL語言可以不須特別的操作系統而寫成。這點我們必須集中注意力。

由於 PASCAL 語言是經過編譯，故此不需要執行每一行，甚至無須理會它在程式上的位置，編譯器同一時間縱觀整個程式，分析出每一個指示的內容。舉例，構想一個大的程式，部份以數字寫成，其中一個變數稱爲〝N〞代表輸入的數目。但在另一程式，可以用〝N〞代表程式編寫員的智商。在APPLESOFT，兩個〝N〞必須是同一價值，因爲APPLESOFT編擇器是懂盲從而不知自己做些甚麽，而 PASCAL，可以用同様的名稱代表兩個不同的變數，只要兩者在程式上不同的區塊裡。

PASCAL區塊每一個都是邏輯上獨立，換言之可以用不同步驟寫成而輸入PASCAL程式內，而無須憂慮它會與程式內其他部份有所衝突干擾。

爲了使邏輯上獨立的功能確實成事，PASCAL 製造新區域以記憶每一次的輸入的步驟及功能。故此 PASCAL 附有記憶系統，而 BASIC 則浪費了這種功能。

PASCAL語言並沒有行數（LINE）。所有步驟及功能都有名稱，又要給予名稱就可給予指示。亦即是說你可以爲語言製造新的命令，呼喚步驟好像命令句字，如果你能適當地給予每個步驟名稱，可以使整個程式易讀易明白。

在 PASCAL，所有變數在應用前都要聲明其代表意思，編譯器必須明白變數是屬於那種類型（整數、數字、字串等等），以便知道須要留多少空間以備進行一個步驟或功能。APPLESOFT 並不須要聲明變數的意義。但PASCAL的好處，可使編譯器避免串錯變數字句或不理智的應用變數。

談到 PASCAL 的程式控制，APPLESOFT 內，〝GOTO〞是用作控制程式流程。用GOTO及ＩＦ能夠使一部分的程式重複使用，如果條件是對的話。

```
10 i=1
20 if i > 64 goto 60
30 print i*i
40 i=2*i
50 goto 20
60 end
```

以上例子顯示完整的APPLESOFT 程式，它印上1，2，4，8，16，32及64的平方，有一個IF及兩個 GOTO，驟眼看來，很不明白這是甚麽，因爲你要指出 GOTO 往那裡。在PASCAL語言，同樣的程式會寫成如下：

```
program squares;
var i:integer;

begin
    i := 1;
    while i <= 64 do
    begin
        writeln(i*i);
        i := 2*i;
    end;
end.
```

在WHILE 後的句子是迴路的主體，WHILE 程式或編寫員提高注意力，比 BASIC 的 IF 及 GOTO 更爲清晰。

兩種語言都有其詞句。在P—ASCAL語言是 BEGIN END 及分號，在 BASIC 語言，則是行號。

除了閱讀準確要化費更大的注意力之外，GOTO 還有另外的缺點：它則是危險，容易產生錯誤。你可能在迴路的中間或其他地方 GOTO 不存在的行綫。單看B—ASIC 程式一部份，你是很難明白其意思，而對 GOTO 是不明所指。而 PASCAL 有足夠的字眼，好像 WHILE 及 IF，令你永遠不需要 GOTO ，並且能夠令部

PASCAL與APPLESOFT的比較

方程式（例如迴路內）與其他部份獨立。如果你能知道那部份的意思，便可繼續下一部，而不用担憂它會與 GOTO 有聯繫。

你可以明白 PASCAL 程式，而不須理會控制上的過程。因爲它讀來好像英語，這是 BASIC 語言絕難做到的。

PASCAL 語言另 個優勝的特點是它的變數可以是任何類型，不局限於數字或字串字母。BOOLEAN 可以有絕對眞假值。你可以用 MARRIED當變數作爲男女婚姻狀況的眞假値，已婚是「對」（TRUE），未婚是「錯」（FALSE）。在 APPLESOFT，這種情況下，就必須要用代碼，變數M的價値1代表已婚，O代表獨身。但這種代碼意義上欠獨立性，當一個月後再看這個時代時，可能會對1，O數字摸不着頭腦，莫名所以，甚至如果程式上出現錯誤，而出現〝2〞，就更加如丈八金剛，但 PASCAL則不會如此，BOOLEAN 變數只會顯示眞假的絕對值。你可以用以下形式寫出程式。

```
type marital--status = (single,
married,divorced,widowed,looking);

var M:marital-status;
```

PASCAL 程式可以清楚明確顯示M代表的婚姻狀況（獨身、已婚、離婚、鰥夫、寡婦、尋伴侶中，比用代碼代表明顯優勝得多），在 BASIC 程式，如果程式員忘記了代碼所代表的意思，就很難說明M＝4 是甚麼婚姻狀況。

上述所說，是 PASCAL程式主要特點，程式容易寫閱，而分爲獨立的單位，意思明確。全部是很有系統及規律與開放性的。

雖然我們經常用 APPLESOFT，但我們仍然推薦用 PASCAL 寫五十行以上的程式，PASCAL 是需要用兩部磁碟機器才能操作得圓滿，但仍比APPLE DOS 磁碟優勝，寫數據爲基礎的程式，遊戲程式需要輸入大量非數字的資訊及要有組織儲存資訊的程式，採用 PASCAL 語言是第一個選擇。

PASCAL 強有力的優點，使它適合應用於重要，涉及大量的時間投資及長期性的不斷重複使用的計劃。APPLESOFT 的彈性及可以配合其他軟件使用的特性，適用於小型計劃及日常的工作上。

PASCAL 與 APPLESOFT/DOS 比較表

操作系統	UCSD PASCAL	APPLESOFT／DOS
	先進而威力大，任何時間螢幕上都可出現選擇指示並可作編輯工作。	DOS 可以採用很多其他軟件，而不能UCSD
程式發展	程式容易閱寫，編輯器幫助進行複雜更改。編譯器可以運行前找出錯誤。	任何更改毋須將整個程式重新編譯，由編輯至編譯沒有時間空間。
程式的維修	程式寫出後，不須再行打理，日後亦容易閱讀，因爲PASCAL是自行文件記錄語言，新加入的密碼變成獨立的單位，不會干擾程式的其他部份。	沒有這條件，BASIC 不善於程式記錄，因爲 REMARK 句子只會浪費空間及使操作速度減慢。
需要的磁碟空間	很多檔案佔用空間，每一程式需要保留兩個檔案空間（一個是文字、一個是密碼）但不能用 APPLE 五吋磁碟。	不很多，因爲BASIC 語言儲存時是濃縮的不需担心大的系統檔案。
語言能力	抽象資訊，記錄程序及功能。	硬件易於使用（PEEK，POKE，CALL）。
操作速度	幾快。	很慢，APPLESOFT 浪費時間於逐行解釋上。
兼容	是準標化的語言，縱然有差別也只少許，一個文字檔案可被編釋及在不同的機器上運作。	所有的 APPLE 機器運作APPLESOFT 及 DOS：所有與 APPLE 相容的產品都支持APPLESOFT及DOS

PASCAL與APPLESOFT的比較

以下列舉一個 APPLESOFT 與 APPLE PASCAL 程式例子執行速度比較。程式是尋找質素。

APPLESOFT 採用的操作系統是DOS，找出由1至1000 的質素需要18秒。而 APPLE PASCAL用的是 UCSD 操作系統，執行同樣的工作只需要10秒。

```
10 REM * PRIME FACTOR *
20 REM
30 REM THIS PROGRAM DETERMINES THE
40 REM PRIME FACTORS OF AN INTEGER
50 REM
70 Z=0 : S=0 : I=0
80 GOTO 4000 :REM MAIN PROGRAM
1000 REM DISPLAY
1010 IF S=0 THEN 1050
1020 PRINT I;
1030 PRINT "^";
1040 PRINT S
1050 RETURN
2000 REM CALCULATE
2005 X=Z
2010 FOR I=2 TO X
2020 S=0
2030 IF (Z/I) = INT (Z/I) THEN Z = Z/I:
S =S+1:GOTO2030
2040 GOSUB 1000: REM DISPLAY
2050 NEXT
2060 RETURN
3000 REM NUM
3010 PRINT
3020 PRINT "NUMBER: ";
3030 INPUT "";Z
3040 IF Z= 0 THEN 3070
3050 GOSUB 2000 :REM CALC
3060 GOTO 3010
3070 RETURN
4000 REM RETURN
4010 HOME
4020 PRINT "PRIME FACTOR ROUTINE"
4030 PRINT "ENTER A ZERO TO END"
4040 GOSUB 3000 :REM NUM
4050 PRINT
4060 PRINT
4070 PRINT "GOODBYE"
4080 END
```

程式（1） BASIC（APPLESOFT）程式

```
PROGRAM PRIME;

(* PRIME FACTOR *)

(* THIS PROGRAM DETERMINES THE *)
(* PRIME FACTOR OF AN INTEGER  *)

VAR
Z   :REAL;
S,I :INTEGER;

PROCEDURE DISPLAY;
 BEGIN
   IF S < > 0
    THEN BEGIN
         WRITE (I);
         WRITE (,^,);
         WRITE LN (S);
       END;
END;

PROCEDURE CALCULATE;
  VAR X : INTEGER;
BEGIN
  X :=TRUNC (Z);
  FOR I :=2 TO X DO
   BEGIN
     S :=0;
     WHILE ((Z/I) =TRUNC (Z/I)) DO
      BEGIN
        Z :=Z/I;
        S :=S+I;
      END;
     DISPLAY;
   END;
END;

PROCEDURE NUM;
 BEGIN
  REPEAT
   WRITE LN;
   WRITE ('NUMBER: ');
   READ  LN(Z);
   CALCULATE;
  UNTIL (Z=0);
END;

BEGIN (* MAIN *)
  PAGE (OUTPUT);
  WRITE LN('PRIME FACTOR ROUTINE');
  WRITE LN('ENTER A ZERO TO END' );
  NUM;

      WRITE LN;
```

程式（2） APPLE PASCAL 程式

蘋果音樂產生原理探討

花生

在機器語言中，聲音的產生在它自己本身來說是一個大題目，可以用整本書來專門討論它。然而，在這裡我們只以簡單的程式來作接觸。這些程式不僅可以使你將已學習過的命令充份應用，更能使你覺得很有意思。

喇叭是使用從$C000 至$ C0FF位置範圍的記憶體的一部份，而這段記憶體是專門保留給 APPLE Ⅱ 的硬體項目使用的。

鍵盤是用$C000的記憶位置，而喇叭是類似的使用$C030的記憶位置，所不同的是在$C000 位置的數值必須依據所按下鍵的資料而變化；然而在 $C030 處却沒有邏輯值被送回。

每一次 $C030的位置被存取，喇叭就會發出卡搭的聲响，這個動作很容易展現。簡單地在 BASIC 中輸入 C-ALL－151 的敍述就會進入監督程式中，再輸入 C030 和按下 RETURN 鍵，你需注意地仔細聽，而且試很多次看看。每一次喇叭都會卡搭地响一次。你能想像，如果我們以足夠的速度重覆地存取時，喇叭將會發出一連串的穩定聲音。在 BASIC 語言中，這個程式亦能被執行，雖然不好，但是像下列的簡單迴路亦是可行的：

```
10X=PEEK（－16336）：GOTO 10
```

在機器語言中，程式如下列所示：

```
0300 — AD 30 C0    LDA    $C030
0303 — 4C 00 03    JMP    $0300
```

要如此做，必須先以 CALL－151 進入監督程式，再輸入

```
300： AD 30  C0 4C 00 03
```

然後再以300 G來執行輸入的程式

程式是正在工作的，可是問題在程式眞實地執行速度太快，以造成喇叭來不及反應的情形。在這裡所缺的是如何控制其迴路的執行速率，這種控制經常是以延遲的迴路方式來處理的，我們亦將在程式中定下其延遲的時間常長，加於程式執行之前或執行之中都可以。

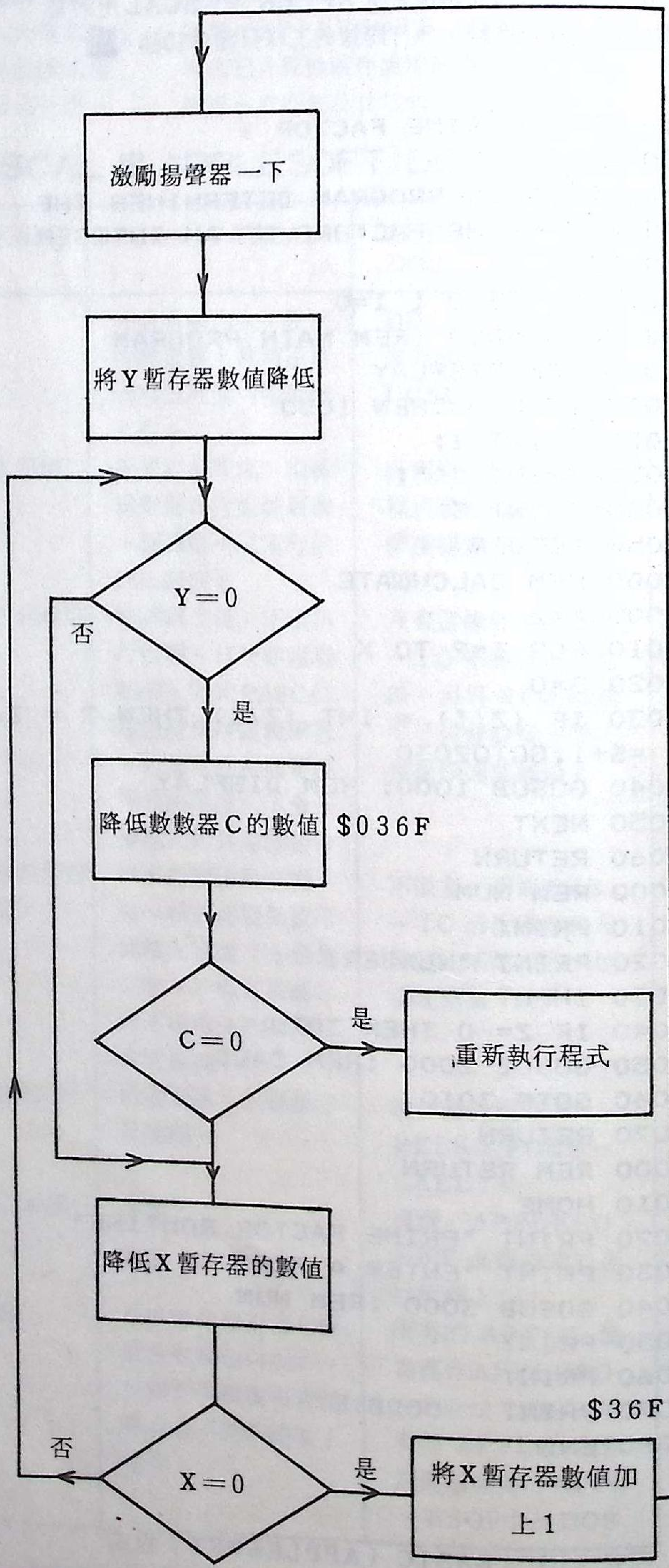
延遲──

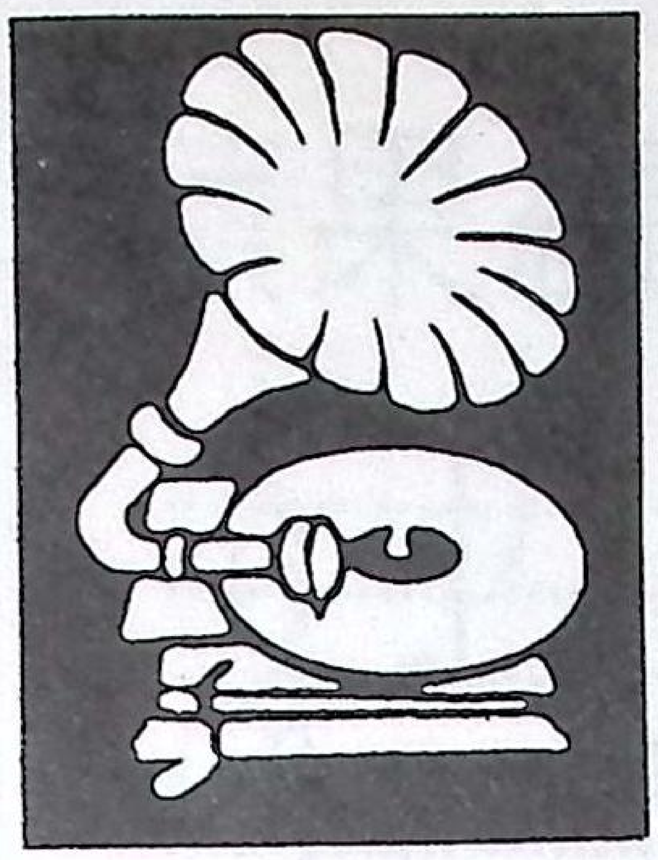
1、實質上改變程式的長度，增加其每次經過迴路的執行時間。

2、在執行程式之前儲存一個數值至記憶體內，再於程式執行中使用這個數值做迴路的延遲。

3、從外界輸入一個數值來做延遲，就像從鍵盤中或歡樂棒輸入一樣。

不過除非是將延遲數值寫入程式中，從鍵盤上直接讀入來控制不同的延遲是不能夠的。一般原理請參考以下流程圖。■





巴哈的古典名作

□黃國祥□

能產生聲音的特點，可以說是 APPLE 電腦受用家擁護的原因之一。讀者們只要細心一想，如果心愛的程序缺少了這個功能，這些程序會變成如何的枯燥缺味。雖然自古常言；〝沉默是金〞，但是在長時期對着不發一言的電腦，用家是何等的煩悶。當然，電腦能產生的聲音音節，是會受到本身客觀條件所影响。本文主要討論的範圍，只有關 AP—PLE 電腦所能產生的聲音。

APPLE 電腦聲音是由佔有記憶位置 49200 的內藏揚聲器所產生的。每當用家接觸到這個記憶區域，揚聲器膜片會受到震動動作，從而產生清楚的音律。讀者們可用以下簡單程序，作一嘗試。

10 FOR X＝1 TO 100：Z＝PEEK（49200）：NEXT X 當然，以此所產生的聲音，是與音樂音符完全不同。故此，爲了克服困難，便引出數年前已開始發展的機械語言。雖然，機械語言部份形式與電腦語言有分別，但大部份程式則仍照一般過程；將需要音符的音節， POKE 入其中一個記存器內，至於其維持長度，則存入另一位置；整個程式，叫做機械語言聲音程式。這種程式的好處，是聲音較準、清楚、記憶效率高及使程式編寫員易於控制。

於 APPLE 電腦上產生音符的標準方法，是有其本身的限制。例如，利用此方法，聲音音量會劃一，而音符長度則爲用家 POKE 入一個簡單記憶（ 255 ）內的最大數目。如強行產生長音符，只會做成混濁效果，與原意剛好相反。此外，程式如翻錄的話，效果亦會不甚理想。要在 AP—PLE 電腦上產生不同音符，一個可行方法，是利用良好音板，借助外加揚聲器，產生不同效果。

由於 APPLE 電腦的內藏揚聲器主要是靠震盪產生聲音。所以，如控制震動的狀態或速度，可達到令人意想不到的效果。例如，用家可控制聲音音量。這技術亦容許同時產生兩個音符及延長音符長度。

筆者嘗試將兩種聲音產生技術介紹出來。第一個程式爲〝標準方法〞，第二個程式將同樣音韻，用〝兩音調技術處理。兩個方法同時列出（列表1），使讀者可作比較。

在音量控制程式中，引入程式如下：

100＆S，X，Y，Z

X表示音符音節，Y表示長度，後者的音符值可達65,535。變數Z表示音量，數值可由0至15。筆者音量從低（1）開始，然後以級數0、1上升。直至完全表達了 142 個音符。

〝兩音調〞技術則用下列引入法：

110＆T，X，Y，Z

變數X表示音節，Y表示長度。至於長度數值，已預校爲單音調程式的兩倍，Z表示第二音符的音質。在這個程式中，筆者選擇第二個音符爲第一個音符音節的一半，因而產生有趣的和音效果。

兩種方法的機械語言程式，貯於記憶位置 784—926 內。□

> 陣列變數：
> - D （*）：音符長度
> - GG（*）：音符音節
> - VP（*）：音符於螢幕上垂直位置
>
> 其它變數：
> - S：通知機械語言程序，用單音符。
> - T：通知機械語言程序，用雙音符。
> - V：音符音量。
> - XX：決定要用的聲音程序。
> - ZA：音符於螢幕上的橫行位置。
> - M，N，ZZ：計數器。

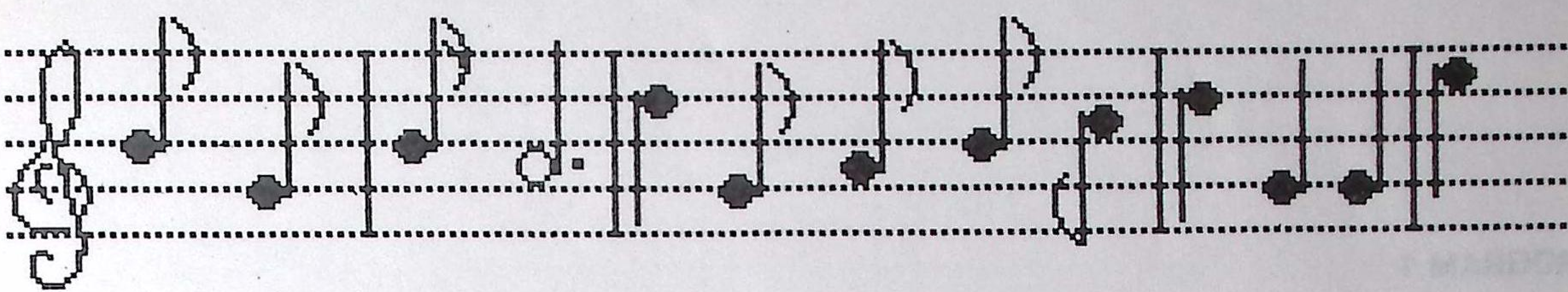
```
]LIST

10  GR : HOME
20  DIM GG(142),VP(108),D(142):V =
     1
30  COLOR= 4: FOR N = 16 TO 32 STEP
     4: HLIN 0,39 AT N: NEXT
40  INVERSE : HTAB 17: PRINT "FUG
    UE": NORMAL
60  FOR ZZ = 784 TO 949: READ N: POKE
     ZZ,N: NEXT
70  POKE 1030,76: POKE 1014,16: POKE
     1015,3
80 VP(33) = 10:VP(35) = 12:VP(37)
     = 12:VP(40) = 14:VP(42) = 1
    4:VP(45) = 16:VP(47) = 18:VP
    (50) = 18:VP(53) = 20:VP(57)
     = 20:VP(60) = 22:VP(64) = 2
    4:VP(68) = 24:VP(72) = 26:VP
    (76) = 26:VP(81) = 28
90 VP(85) = 28:VP(91) = 30:VP(96)
     = 32:VP(102) = 32:VP(108) =
     34
```

```
100  FOR ZZ = 1 TO 142: READ N:GG
     (ZZ) = N: READ M:D(ZZ) = M
110  NEXT
120  IF XX = 0 THEN  VTAB 23: CALL
     - 958: PRINT " MUSIC REPROD
     UCTION BY COMPUHOUSE"
130  IF XX = 1 THEN  VTAB 23: CALL
     - 958: PRINT "  SINGLE TONE
      SOUND WITH VOLUME"
140  IF XX = 2 THEN  VTAB 23: CALL
     - 958: PRINT "  DUAL TONE S
     OUND REPRODUCTION"
200 ZA = 0: FOR N = 1 TO 142
210  IF GG(N) = 0 THEN  FOR M = 1
      TO 400: NEXT : GOTO 290
220 ZA = ZA + 2: IF ZA > 39 THEN
     ZA = 2
240  IF XX = 0 THEN  POKE 927,GG(
     N): POKE 928,D(N) * 40: CALL
     929: GOTO 270
250  IF XX = 2 THEN  & TGG(N),200
     0 * D(N),GG(N) / 2: GOTO 270

260  IF XX = 1 THEN V = V + .1: &
     SGG(N),1000 * D(N),V
270  COLOR= 0:VP = VP(GG(N)): IF
     VP = 16 OR VP = 20 OR VP = 2
     4 OR VP = 28 OR VP = 32 THEN
      COLOR= 4
280  PLOT ZA,VP
290  NEXT : IF XX = 0 THEN D(142)
      = 26
300  IF XX = 2 THEN D(142) = 6
310 XX = XX + 1: IF XX > 0 AND XX
      < 3 THEN  FOR M = 1 TO 3000
     : NEXT : GOTO 120
320 XX = 0:V = 1: VTAB 23: CALL  -
     958: FOR N = 1 TO 2000: NEXT
     : GOTO 120
1000  DATA  201,83,240,7,201,84,2
     40,55,76,201,222,32,132,3,13
     8,41,15,133,255,165,253,133,
     251,198,251,208,25,173,48,19
     2,165
1010  DATA  253,133,251,164,255,2
     34,234,136,16,251,173,48,192
     ,164,255,234,200,192,16,144,
     250,160,10,136,208,253,32,14
     3,3,144,217,96
1020  DATA  32,132,3,134,254,165,
     253,133,251,165,254,133,252,
     198,251,240,6,70,255,165,255
     ,16,7,173,48,192,165,253,133
     ,251,198,252,240,6,70,255,16
     5,255,16,7,173,48,192,165
1030  DATA  254,133,252,32,143,3,
     144,217,96,32,245,230,134,25
     3,32,190,222,76,70,231,165,8
     0,208,8,165,81,208,2,56,96,1
     98,81,198,80,24,96
1040  DATA  0,0,173,48,192,136,20
     8,5,206,160,3,240,9,202,208,
     245,174,159,3,76,161,3,96
1050  DATA  81,4,53,4,68,6,72,2,8
     1,2,68,2,72,2,81,2,85,2,72,2
     ,108,4
1060  DATA   81,2,108,2,72,2,108,
     2                  ,68,2,72,1,81,1
     ,72,2,108,2,81,              2
     ,108,1,81,1,72,2,108,1,72,1
                      ,68,2,72,1,81,1,7
     2,1,108,1,5                  3,1
1070  DATA  60,1,68,1,72,1,81,1,6
     8,1,72,1,81,1,85,1,72,1,81,1
     ,108,1,81,1,72,1,68,1,60,1,5
     3,1,47,1,45,1,47,1,53,1,45,1
     ,47,1,53,1,57,1,47,1,53,2
1080  DATA  72,2,53,2,47,2,45,1,4
     0,1,45,1,40,1,45,1,40,1,45,1
     ,40,1,35,1,40,1,35,1,33,1,35
     ,1,40,1,45,1,47,1,45,1,35,1,
     40,1,35,1,57,1,35,1,40,1,35,
     1,53,1,35,1,40,1,35,1,57,1,3
     5,1,40,1,35,1
1090  DATA  45,1,53,1,57,1,53,1,4
     0,1,53,1,57,1,53,1,35,1,53,1
     ,57,1,53,1,40,1,53,1,57,1,53
     ,1,72,2,45,2,81,2,47,2,91,2,
     72,2,53,2,45,2
1100  DATA  50,2,35,2,0,2,50,2,53
     ,2,40,2,0,2,53,2,60,1,68,1,6
     0,1,53,1,60,1,35,1,40,1,35,1
     ,68,1,40,1,42,1,40,1,72,4,45
     ,4,81,4,47,4,91,6
```

蘋果編曲器



在未讀本段前，請讀者應先輸入程式1及將它存貯好在電腦內。然後，運行程式。如果輸入正確的話，讀者可見螢幕頂行有 SOUND ＃1及隨後的一小段序言。

一般而言，讀者們從 APPLE 電腦中聽到的聲音應爲鐘聲。而這種聲音，可按下 CTRL 鍵及G鍵，使其立卽再生。在程式 SOUND #1中，讀者們應發現到行號 30 內的 CHR$（7），印出值爲一7，這數值其實是 CTRL－G 的數字代碼。如果讀者正在使用 INTEGER BASIC，就要利用行號 35 的格式。在這行內，讀者可見 PRINT 之後，有兩個並排引號。在這兩個引號內，爲 CTRL－G。至於行號 37 的REM 句子，則表示出如何打入行號 35。有一點應要記得，任何句子的控制性能絕對不可能在行列出或打入它們時，可以預先明白得到。不過，唯一有趣的副作用，是在利用行號 35 的方法列出程式時，讀者們應能聽到程式內的所有鐘聲。

在程式1中，電腦是等待着用者按下鍵掣。如果按下R鍵，就會將用以上程式行號產生的任何聲音重覆。如果按下C掣，就會繼續移至程式1內的下一聲音。任何的其它鍵掣（除了 RESET 外），則不會產生任何作用。

"的滴"——16336

現在，可按下鍵掣C，使進入程式 SOUND ＃2。在這個程式內，有一個簡單的 FOR／NEXT 迴路來使APPLE的揚聲器鳴聲，鳴叫十次。留意行號 80 末端的分號。它的使用是防止螢幕上捲，如果不使用分號，每當C－TRL－G 被印出時，游標會向螢幕下移直至螢幕開始向上捲。

現在，讀者請留意 SOUND ＃3。可見到－16336，其實它是 APPLE 揚聲器的記憶地址。每次當這地址被存取，APPLE 便會給予揚聲器輕微推動，從而產生細小"的搭"聲。利用行號130的 PEEK，是一個存取地址的簡單方法，如果第一次錯過聲音，可按下R鍵再聽一次。

SOUND＃4包括另一簡單迴路，它會將揚聲器的記憶地址，PEEK 100 次。將 −16336 編入變數 NO（代表 NOISE），可代替在希望利用它時，每次都要打入 — 16336。當然，用者可隨自己喜愛，用任何變數。

在 SOUND＃5 內，讀者應看見到行號 250將很多"的搭"聲串起。這做法會產生較 SOUND＃ 3爲長的聲音及較 SOUND＃4 的高音的聲音。有一點讀者應要知道，愈頻密的 PEEK，聲音音調愈高。行號 250 中，讀者會留意到共 PEEK－16336 達15 次。

最後，SOUND＃6 示範了有關"的搭"。它用一個 FOR／NEXT 迴路使行號 320重覆100次。此外，它有正及負符號表示出與前面一般常用的不同。

控制桿

程式2需用到控制桿歡樂棒。它是一個簡單的BASIC程式，將 DATA 句子中的 BYTE 讀出及 POKE 入記憶位置768（$300）至786（$312）。程式開端是先 CALL 768，如果程式輸入正確的話，讀者可聽到高音鳴聲；但當轉動歡樂棒，鳴聲音調會轉變。要離開程式，可按下 RESET 或 CTRL－RESET，不過方法要視機型而定。

程式3爲這個小型機械語言程式源碼。下面爲程式的簡略解釋：

（1）將控制桿位置存入X記存器。

（2）跳至 PREAD 子程式（可見 APPLE Ⅱ 參考手册），PREAD 作爲一個延遲使用，但視控制桿位置而定。

（3）利用到達－16336（$C 030）區域，接連急迫揚聲器。

（4）重覆下一控制桿。

（5）跳至開端。

由於聲音音調靠揚聲器扯動次數的密度而定，故控制桿調定愈低，音調則愈高。

音樂製造

現在，可以討論在 APPLE 中用來產生音符的程式。當然，程式是有一定限制，例如，它不能產生有鼓伴音的完整貝多芬交響樂曲。但假如讀者們覺得有此需要，可利用其它特別的周邊設備來造到。不過，無論如何，用一般的 A－PPLE 硬件設備，已足夠作出多樣變化了。

程式4是一個簡單的 BASIC 程式，用來 POKE 入機械語言子程式，成立部份參數，及 CALL 子程式等。程式本身會繼續運行直至按下鍵掣。讀者們可嘗試運行。如果從未聽過 APPLE 音符，可能會對此產生特別興趣。

在程式 POKE 入子程式後，它 POKE 隨機數（音調）入位置 768（$300） 及POKE 一個隨機數（持續期）入 769（$301）。能 POKE 入這位置最大值爲255。

程式5爲"音符產生"程式的源碼，它 POKE 入程式4的記憶內。其實，程式工作頗似"控制桿聲音"。最大分別是，位置768及 769控制音程持續期及音調，來代替了控

蘋果編曲器

PROGRAM 1

```
10 I = 10: HOME
20  PRINT "SOUND #1": PRINT : LIST
   30,37
30  PRINT  CHR$ (7)
35  PRINT "": REM CTRL-G
37  REM  ?"CTRL-G"
40  GOTO 10000
50 I = 50: HOME
60  PRINT "SOUND #2": PRINT : LIST
   70,90
70  FOR LOOP = 1 TO 10
80  PRINT  CHR$ (7);
90  NEXT
100  GOTO 10000
110 I = 110: HOME
120  PRINT "SOUND #3": PRINT : LIST
   130
130 X =  PEEK ( - 16336)
140  GOTO 10000
150 I = 150: HOME
160  PRINT "SOUND #4": PRINT : LIST
   170,200
170 NO =  - 16336
180  FOR LOOP = 1 TO 100
190 X =  PEEK (NO)
200  NEXT
210  GOTO 10000
220 I = 220: HOME
230  PRINT "SOUND #5": PRINT : LIST
   240,260
240 NO =  - 16336
250 X =  PEEK (NO) +  PEEK (NO) +
    PEEK (NO) +  PEEK (NO) +  PEEK
   (NO) +  PEEK (NO) +  PEEK (N
   O) +  PEEK (NO) +  PEEK (NO)
    +  PEEK (NO) +  PEEK (NO) +
    PEEK (NO) +  PEEK (NO) +  PEEK
   (NO) +  PEEK (NO)
260  REM  FIFTEEN TIMES
270  GOTO 10000
280 I = 280: HOME
290  PRINT "SOUND #6": PRINT : LIST
   300,330
300 NO =  - 16336
310  FOR LOOP = 1 TO 100
320 X =  PEEK (NO) -  PEEK (NO) +
    PEEK (NO) -  PEEK (NO) +  PEEK
   (NO) -  PEEK (NO) +  PEEK (N
   O)
330  NEXT
10000  POKE  - 16368,0: VTAB 20: HTAB
   1: CALL  - 958: PRINT "'R' F
   OR REPEAT, 'C' TO CONTINUE "
   ;: GET A$
10010  IF A$ <  > "R" AND A$ <  >
   "C" THEN 10000
10020  IF A$ = "C" THEN 10100
10030  IF I = 10 THEN 30
10040  IF I = 50 THEN 70
10050  IF I = 110 THEN X =  PEEK
   ( - 16336): GOTO 130
10060  IF I = 150 THEN 170
10070  IF I = 220 THEN 240
10080  IF I = 280 THEN 300
10100  IF I = 10 THEN 50
10110  IF I = 50 THEN 110
10120  IF I = 110 THEN 150
10130  IF I = 150 THEN 220
10140  IF I = 220 THEN 280
10150  TEXT : HTAB 1: PRINT "END
   OF PROGRAM#1"
```

PROGRAM 2

```
10  REM  PROGRAM#2
20  FOR LOC = 768 TO 786: READ BY
   TE: POKE LOC,BYTE: NEXT LOC
30  DATA  162,0,32,30,251,141,48,
   192,162,1,32,30,251,141,48,1
   92,76,0,3
40  CALL 768
```

```
*300L

0300-    A2 00       LDX   #$00
0302-    20 1E FB    JSR   $FB1E
0305-    8D 30 C0    STA   $C030
0308-    A2 01       LDX   #$01
030A-    20 1E FB    JSR   $FB1E
030D-    8D 30 C0    STA   $C030
0310-    4C 00 03    JMP   $0300
0313-    8A          TXA
0314-    20 C1 FB    JSR   $FBC1
0317-    A0 27       LDY   #$27
0319-    B1 28       LDA   ($28),Y
031B-    91 06       STA   ($06),Y
031D-    88          DEY
031E-    10 F9       BPL   $0319
0320-    E0 00       CPX   #$00
0322-    F0 02       BEQ   $0326
0324-    10 E4       BPL   $030A
0326-    A0 27       LDY   #$27
0328-    A9 A0       LDA   #$A0
032A-    91 28       STA   ($28),Y
```

PROGRAM 3

PROGRAM 4

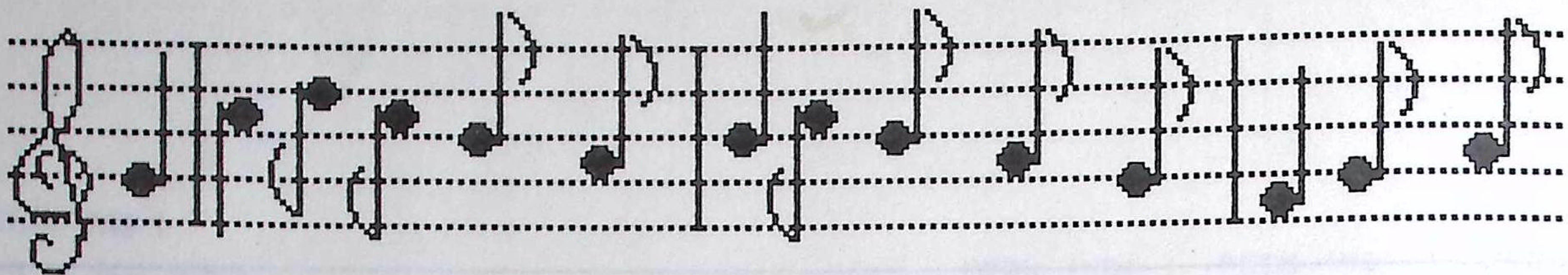
```
]LIST

5  REM  PROGRAM#4
10  FOR LOC = 770 TO 790: READ BY
    TE: POKE LOC,BYTE: NEXT
20  POKE 768, INT ( RND (1) * 255
    ) + 1: POKE 769, INT ( RND (
    1) * 100) + 1: CALL 770:X =
    PEEK ( - 16384): IF X < 127
    THEN  POKE  - 16368,0: GOTO
    20
30  DATA  173,48,192,136,208,5,20
    6,1,3,240,9,202,208,245,174,
    0,3,76,2,3,96
40  POKE  - 16368,0
```

PROGRAM 5

```
*300L

0300-    E6 00       INC   $00
0302-    AD 30 C0    LDA   $C030
0305-    88          DEY
0306-    D0 05       BNE   $030D
0308-    CE 01 03    DEC   $0301
030B-    F0 09       BEQ   $0316
030D-    CA          DEX
030E-    D0 F5       BNE   $0305
0310-    AE 00 03    LDX   $0300
0313-    4C 02 03    JMP   $0302
0316-    60          RTS
0317-    A0 27       LDY   #$27
0319-    B1 28       LDA   ($28),Y
031B-    91 06       STA   ($06),Y
031D-    88          DEY
031E-    10 F9       BPL   $0319
0320-    E0 00       CPX   #$00
0322-    F0 02       BEQ   $0326
0324-    10 E4       BPL   $030A
0326-    A0 27       LDY   #$27
```



制桿控制聲音音調。源碼包括一些備註資料，使讀者更加明瞭。

至此，讀者應明白每當用到聲音程式，應用位置 — 16336（$C030）。讀者可嘗試用程式5來 POKE 入喜愛的音符值及聽取效果。

以下是介紹一個編曲器程式給大家。

編曲器程式

〝SONG WRITER程式能令用家隨自己所好，輕易地編出他們所喜愛的歌曲。當用家將程式運行時，螢幕首先所見，是程式之標題，然後，便會聽到部份可能熟悉的音調。在音調停頓後，螢幕會顯示 COMMAND?及閃動的游標，在螢幕上端，用家可見一系列的可用指令及相對字母。在螢幕右邊，則可見各音符名稱及相對的值。

最重要一件事，是用家要明白及清楚如何使用指令。所以，筆者會在此加以詳細討論，並順螢幕所示之次序。這些指令均只適合回答 COMMAND？。在每次輸入指令後，可按 RETURN鍵。筆者提議讀者們，應在閱讀本文時，同時進行實習。

指令

A＝ADDNOTE

這指令會使用家打開音樂檔案（歌曲）及保持新近加入的，每當用家按A鍵（及 RETURN）便會先輸入音符，逗號，及持續期。例如：

NOTE＃1

NOTE，DURATION 128，200

最大音符值是 255（其實0＝256），這原理同樣適用於持續期值。當在輸入各數值後，用家會聽到新的音符實際的聲音。

E＝EDIT

當用家不幸出錯，可打入E（當然，亦要用 RETURN）來將它固定。稍後，螢幕會顯示並問用家所希望編輯的音符號數。如果所要編輯的音符並不是音樂檔案的一部份，用家會被詢問音符號數。假如輸入音符值正確的話，舊值會隨即顯示，和即以新值代替。這規則同樣亦適用於在 ADDNOTE指令下輸入資料。例如用家想用新值 64 及 200代替舊值音符1，其做法如下：

COMMAND？E

EDIT NOTE＃1

NOTE＃1 OLO：NOTE＝128 DUR＝200

NOTE，DURATION W＝64，200

P＝PLAY

打入P會將電腦進入重播狀態。這會將歌曲重播及同時在螢幕上顯示出來。由於電腦要在同一時間列出及運行音樂檔案，故運行速度與程式本身不同。它會較慢及顯著。在輸入後，用家會被要求定出起始及結尾之音符。如果不依指示去做，而只按 RETURN ，整首歌會被播出。

S＝SAVE

這指令會貯存（ SAVE）音樂檔案至磁碟。首先，用家會被要求打入檔案名稱，這名稱是將來檔案要存入磁碟時所需用到的。然後，用家要決定存入磁碟的頭及尾音符數目。下一問題是 FOR FUTURE ADDITION?這裡，要略作解釋一下。一般而言，共有兩種可用此指令產生的兩類檔案。如果對以上問題答案爲Y，新檔案會成立並能在任何時間再負載入 SONG WRITER內。用家們一定要利用這特點，以備將來加入任何音符或編輯歌曲。如果答案爲N，成立的檔案很易被轉爲 BASIC 程式，並在運行時，即能重播歌曲。

如果在答 FUTURE ADDITION? 問題時答案爲N，用用家就會被查問未來的 BASIC 音樂程式之開始行號數。然後，電腦便繼續問用家是否希望一個 FULL LOADER PROGRAM ?假如答案是Y，所創造的 BASIC 程式會包括所需資料，以便新的檔案運行時，機械語言〝音符產生”（〝NOTE PRODUCER”）（見上文）程式會被 POKE 入。但是假如答案是N的話，這程式會不包括在內。所以，在程式本身已有某種〝音符”（〝NOTE”）程式，就可答N。

特別要點

最後，電腦會要求用家檢查錯誤。如所有正確無悮，輸入Y後，檔案便被貯存。如輸入N，用家就要重覆整個貯存過程。至此舉例如下：

COMMAND？S

（清除螢幕畫面）

FILENAME？SONG 1

STARTING NOTE NUMBER 2

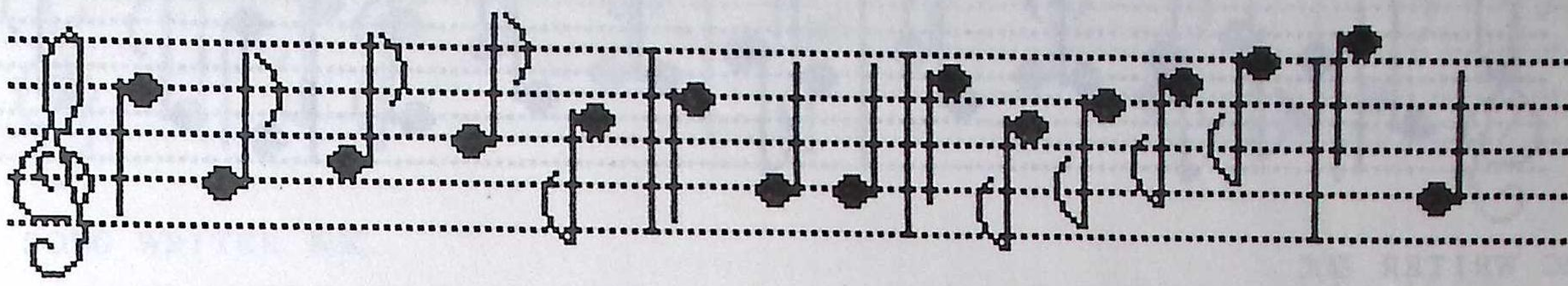
ENDING NOTE NUMBER：10

FOR FUTURE ADDITION ？N

STARTING LINE NUMBER：100

FULL LOADER PROGRAM？Y

IS EVERYTHING OK？Y



這時，音樂檔案會被貯存入檔案名稱 SONG 1 之內。檔案包括由2至10的音符，而所產生的程式會在行號100作始。另一方面，產生的程式已包括了機械語言〝音符〞（〝NOTE〞）程式。

L＝LOAD

如果在答 SAVE 指令下的 FOR FUTURE ADD－ITION？問題時，答案爲Y。用家便能再 LOAD一個舊音樂檔案入電腦。不過，這時，以前所入的資料會消失，答案應用N。只需輸入適當檔案名稱，用家便能處理或再次加入資料。

N＝NORPLAY

本文較早時曾經提及，當重播（PLAY／LIST）歌曲時，播出速度會較正常爲慢，因爲，電腦要同時列出及播出音符。要解決問題，可用 NORPLAY 指令（這指令全寫爲 NORMAL PLAY）；用家們不妨一試。

D＝DELETE

在 COMMAND ？狀態時，如輸入D。電腦便發問究竟要消去那個音符。如果只打入 RETURN 而不理會問題，音樂檔案的最後音符會被刪去。如果對第一個問題答覆爲一個數值時，用家就會被詢問從此音符計起總共要

存貯——最好在存貯時共存入兩個相同版本的音樂檔案入磁碟。其一應在 FUTURE ADDITION?時就要造出來。以便日後可編輯或加入任何資料。如果有需要，另外一個應在程式創造或 FUTURE ADDITION ？N狀態時完成。要留意一點。兩個程式要用不同檔案名稱。

拍子速度——在輸入音符持續期，要記着如果四份一音符之長度爲50 時，半拍音符值便爲 100 ，其它依比

消去的音符數目。然後，適當音符數目會消去。再次螢幕打出 COMMAND？

I＝ INSERT

這指令剛好與 DELETE 相反。對這問題而言，只需根據自己既定的輸入數目，插入適當音符便可。

R＝RESTART

這指令將過程重新開始，由音符1起。

C＝CATALOG

這指令會取到標準 DOS 目錄。

Q＝QUIT

用這指令能將程式單獨取出，換言之，資料完全沒有存入磁碟內。如果偶不小心，用 GOTO 200可再將程式輸入而沒有損失磁碟。

·＝DOS

這指令表示在一般DOS 指令後打入完整指令，會運行這指令。普通用法如下：

COMMAND？·DELETE FILENAME注意一點：某些DOS 指令會使 SONG WRITER 夫去功能，繼使資料損失，請讀者們留意。

H＝HARD

如果電腦有印字機接連，在 COMMAND？ 指令後打入H，便可印出整個列表。注意，用家可能要針對個別印字機，修改行號1210及1220。

類推。其實，用家本身要決定某些特別音符的長度，至於所餘下的音符值，可用1。

數目限制——一首歌曲的音符數目是有限制的。本文程式中只定下數爲 500 ，但改變行號 120 的L值，可增加限額。

音符——螢幕上的音符列出對將樂譜轉入磁碟，頗爲方便。列出數目應爲中音階，要求出高音階，將數目除2，要求出低音階，將數目乘2。例如，音符F用數目36，72 及 144 表示，讀者們更可自行製表列出所有音符及數目。

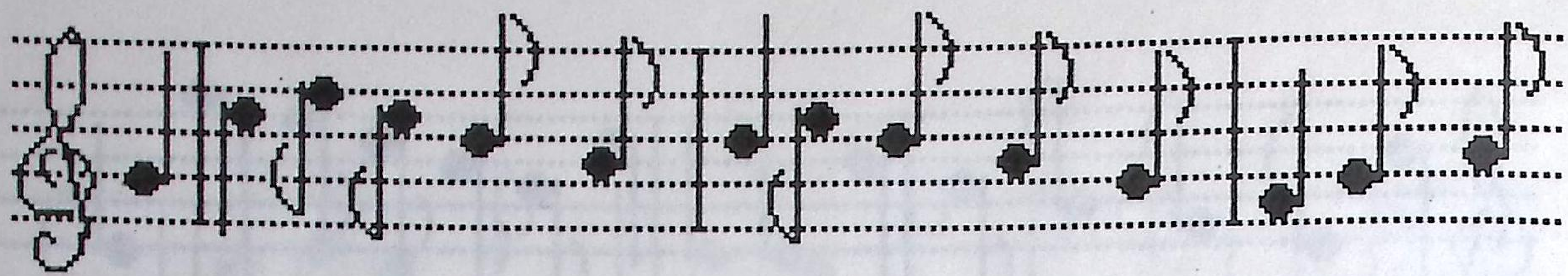
記得一點，F＃與G－FLAT 相同。此外，數目0相等於數目 256。

運行——要使用在 FUTURE ADDITION ？N狀態的所產生的程式，可將它通過 EXEC運行， EXEC是一個DOS 指令，它將編輯檔案在螢幕上印出。故 此用家可運行及將它當作 BASIC 程式處理。最後，更 可將它貯存。另外一個特點，用家可 LOAD 舊BASIC 程式及運行入聲音 程式。要用得恰當，應在存入音樂檔案 時，指定起始行數目，以免音樂程式行數與新加入程式的 子程式產生衝突。

插入—如果有大量重覆性資料要輸入，最好先打入資料中之最後音符，然後才插入其它，這做法可減少在C－OMMAND？狀態下重覆輸入A。

鍵掣——另外，要清楚鍵掣的用途，在畫面混亂的情況下，可利用 ESC－SHIFT－P，此外，CATAL－OG或PLAY／LIST等，可隨時利用 CTRL－S 來停止及按任何鍵掣令其再行工作。最後，讀者們應留意CTRL－C RETURN是有用但可能產生危險的指令，筆者在此只推薦使用 CTRL－C 指令。

實習——不論任何冗長，如不加以實習，是很難掌握其奧妙之處。在未做任何程序前，應先清楚其來龍去脈。才好着手進行。■



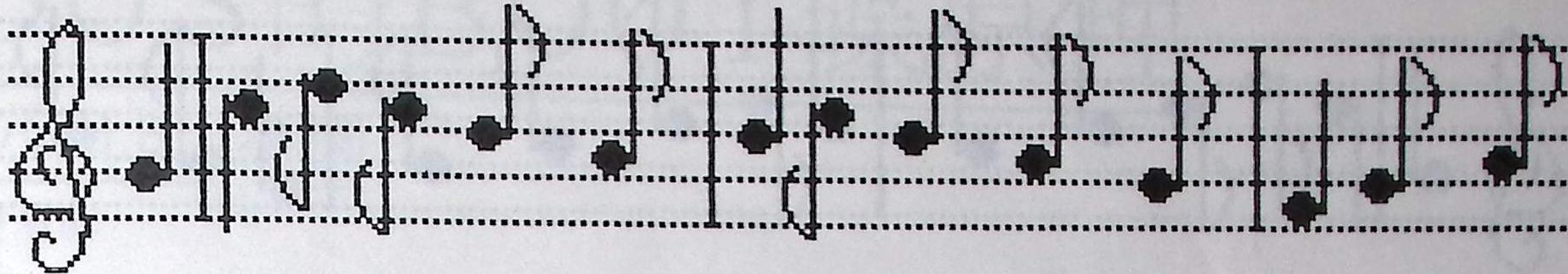
SONG WRITER 程式

```
10  REM     APPLE MUSIC WRITER
20  REM  INITIALIZATION
30  TEXT : HOME : VTAB 1: PRINT "
     A=ADDNOTE E =EDIT   P=PLAY
     S=SAVE   L=LOAD   N=NORPLAY
     D=DELETE  I=INSERT  R=RESTA
     RT C=CATALOG Q=QUIT .=DOS H=
     HARD": PRINT "--------------
     ---------------------------":
     POKE 34,5
40  VTAB 6: HTAB 34: PRINT "G =64
    ": PRINT  TAB( 34)"F#=68": PRINT
     TAB( 34)" F =72": PRINT  TAB(
    34)"E = 76": PRINT  TAB( 34)
    "D#=81": PRINT  TAB( 34)" D
    =86": PRINT  TAB( 34)"C#=91"

50  PRINT  TAB( 34)"C =96": PRINT
     TAB( 34)"B =102": PRINT TAB(
    34)"A#=108": PRINT  TAB( 34)
    "A =115": PRINT  TAB( 34)"G#
    =121": PRINT  TAB( 34)"G =12
    8": PRINT  TAB( 34)"/2 FOR":
     PRINT  TAB( 34)"HIGHER": PRINT
     TAB( 34)"*2 FOR": PRINT  TAB(
    34)"LOWER": POKE 33,32
60  FOR LOC = 770 TO 790: READ BY
    TE: POKE LOC,BYTE: NEXT
70  DATA  173,48,192,136,208,5,20
    6,1,3,240,9,202,208,245,174,
    0,3,76,2,3,96
80  HOME : INVERSE : VTAB 10: HTAB
    9: PRINT "APPLE MUSIC WRITER
    "
90  FOR R = 1 TO 26: READ P,D: POKE
    768,P: POKE 769,D: CALL 770:
     NEXT R
100  DATA  172,75,162,75,152,75,1
    44,75,108,100,1,30,144,75,10
    8,100,1,30,144,75,108,255,1,
    10,108,75,96,75,91,75,86,75,
    108,75,96,75,86,100
110  DATA  1,10,115,75,96,100,1,1
    0,108,150,144,150,216,200,
120  HOME :L = 500: DIM N(L),D(L)
    ,N$(L),D$(L),NN(L),ND(L)
130  REM  MAIN ROUTINES START
140  VTAB 5: GOTO 190
150 I = I + 1
160  PRINT : INVERSE : PRINT "NOT
    E#"I: NORMAL
165  INPUT "NOTE,DURATION ";N$(I)
    ,D$(I): IF N$ =  CHR$ (8) OR
    D$ =  CHR$ (8) THEN N$(I) =
    N$(I - 1):D$(I) = D$(I - 1)
170 N(I) =  VAL (N$(I)):D(I) =  VAL
    (D$(I)): IF N(I) > 255 OR N(
    I) < 0 OR D(I) > 255 OR D(I)
     < 0 THEN 160
180  POKE 768,N(I): POKE 769,D(I)
    : CALL 770
190  ONERR  GOTO 370
200  PRINT : INPUT "COMMAND? ";A$

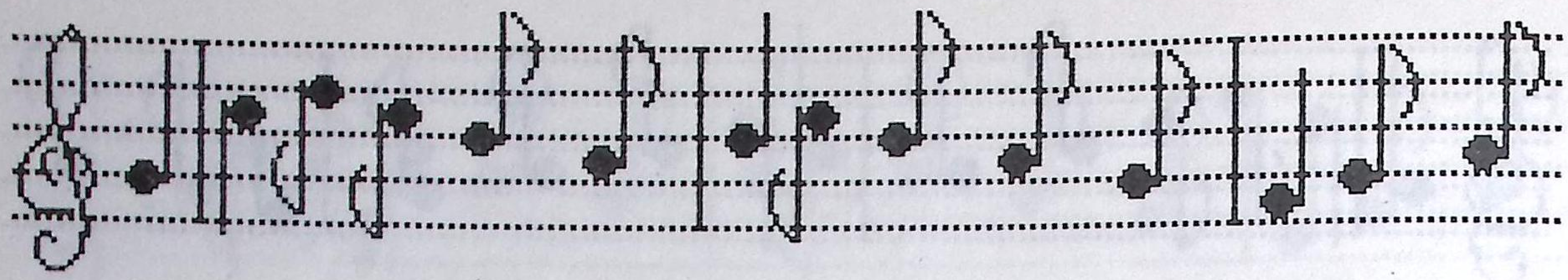
210  IF A$ = "A" AND I = L THEN  PRINT
    "YOU ARE AT YOUR LIMIT!!!": GOTO
    200
220  IF A$ = "A" THEN 150
230  IF I <  = 0 AND (A$ = "E" OR
    A$ = "P" OR A$ = "H" OR A$ =
    "N" OR A$ = "I" OR A$ = "S")
     THEN  PRINT "SORRY, NO NOTE
    S":I = 0: GOTO 190
240  IF A$ = "Q" THEN 450
250  IF A$ = "E" THEN 470
260  IF A$ = "P" THEN 390
270  IF A$ = "S" THEN 530
280  IF A$ = "D" THEN 1410
290  IF A$ = "L" THEN 990
300  IF A$ = "R" THEN I = 0
310  IF A$ = "C" THEN  PRINT  CHR$
    (4)" CATALOG"
320  IF  LEFT$ (A$,1) = "." THEN
    1120
330  IF A$ = "H" THEN 1160
340  IF A$ = "N" THEN 1250
350  IF A$ = "I" THEN 1310
360  GOTO 190
370  PRINT "ERROR#" PEEK (222): GOTO
    190
380  REM  PLAY ROUTINE
390  PRINT : INPUT "STARTING NOTE
    (D=1): ";SN$:SN =  VAL (SN$
    ): IF SN$ = "" THEN SN = 1
400  PRINT : INPUT "ENDING NOTE
    (D=LAST): ";EN$:EN =  VAL (E
    N$): IF EN$ = "" THEN EN = I

410  IF SN < 1 OR SN > I OR EN <
    1 OR EN > I THEN 390
420  PRINT : INVERSE : PRINT "STA
    RT OF SONG": PRINT : NORMAL
    : FOR X = SN TO EN: POKE 768
```



SONG WRITER 程式

```
     ,N(X): POKE 769,D(X): PRINT
     "NOTE#";X;: HTAB 10: PRINT "
     NOTE=";N(X);: HTAB 19: PRINT
     "DURATION=";D(X): CALL 770: NEXT
     X
430  INVERSE : PRINT : PRINT "END
      OF SONG": NORMAL
440  GOTO 190
450  TEXT : HOME : PRINT "GOODBYE
     ": END
460  REM  EDIT ROUTINE
470  INPUT "EDIT NOTE# ";NN: IF N
     N > I OR NN < 1 THEN 470
480  PRINT : INVERSE : PRINT "NOT
     E#"NN;: NORMAL : PRINT " OLD
     : NOTE="N(NN);" DUR="D(NN)"
490  INPUT "NOTE,DURATION: ";N$(N
     N),D$(NN):N(NN) =  VAL (N$(N
     N)):D(NN) =  VAL (D$(NN)): IF
     N(NN) > 255 OR N(NN) < 0 OR
     D(NN) > 255 OR D(NN) < 0 THEN
     480
500  POKE 768,N(NN): POKE 769,D(N
     N): CALL 770
510  GOTO 190
520  REM  SAVE ROUTINE
530  ONERR  GOTO 860
540  HOME : INPUT "FILENAME? ";FI
     $: IF FI$ = "" THEN 540
550  PRINT : INPUT "STARTING NOTE
      NUMBER: ";SN: IF SN < 1 OR
     SN > I THEN 550
560  PRINT : INPUT "ENDING NOTE N
     UMBER: ";EN: IF EN > I OR EN
      < 1 THEN 560
570  PRINT : INPUT "FOR FUTURE AD
     DITION? ";A$: IF A$ <  > "N"
      AND A$ <  > "Y" THEN 570
580  IF A$ = "Y" THEN  POKE 216,0
     :F2 = 1: GOTO 640
590 F2 = 0
600  PRINT : INPUT "STARTING LINE
     NUMBER : ";SL: IF SL > 63900
      OR SL < 0 THEN 600
610  PRINT : INPUT "FULL LOADER P
     ROGRAM? ";A$:A$ =  LEFT$ (A$
     ,1): IF A$ <  > "Y" AND A$ <
      > "N" THEN 610
620  IF A$ = "Y" THEN FL = 1
630  IF A$ = "N" THEN FL = 0
640  PRINT : INPUT "IS EVERYTHING
      OK? ";A$: IF  LEFT$ (A$,1) =
     "Y" AND F2 = 1 THEN 880
650  IF  LEFT$ (A$,1) = "Y" AND F
     2 <  > 1 THEN 670
660  GOTO 190
670 D$ =  CHR$ (4): PRINT D$"OPEN
     "FI$
680  PRINT D$"DELETE"FI$
690  PRINT D$"OPEN"FI$
700  PRINT D$"WRITE"FI$
710  IF FL <  > 1 THEN  GOTO 740
720  PRINT SL;"FORLOC=770TO790:RE
     ADBYTE:POKELOC,BYTE:NEXT":SL
      = SL + 2
730  PRINT SL;"DATA173,48,192,136
     ,208,5,206,1,3,240,9,202,208
     ,245,174,0,3,76,2,3,96":SL =
     SL + 2
740  PRINT SL;"FORR=1TO";EN - SN +
     1;":READP,D:POKE768,P:POKE76
     9,D:CALL770:NEXTR":SL = SL +
     2
750  FOR Z = SN TO EN
760 N = N + 1: IF N = 20 THEN N =
     1
770  IF N <  > 1 THEN 810
780  PRINT
790  PRINT SL;"DATA";
800 SL = SL + 2
810  PRINT N(Z);",";D(Z);: IF N <
      > 19 THEN  PRINT ",";
820  NEXT Z
830  PRINT
840  PRINT D$"CLOSE"
850  GOTO 190
860  PRINT : PRINT  CHR$ (7);"ERR
     OR#"; PEEK (222): PRINT D$"C
     LOSE": GOTO 190
870  REM  2ND SAVE ROUTINE
880  ONERR  GOTO 980
890 D$ =  CHR$ (4): PRINT D$"OPEN
     "FI$
900  PRINT D$"DELETE"FI$
910  PRINT D$"OPEN"FI$
920  PRINT D$"WRITE"FI$
930  FOR S = SN TO EN
940  PRINT N(S): PRINT D(S)
950  NEXT S
960  PRINT D$"CLOSE"
970  GOTO 190
980  REM  LOAD ROUTINE
990  ONERR  GOTO 1090
1000  INPUT "YOU WILL LOSE YOUR D
     ATA, OK? ";OK$:OK$ =  LEFT$
     (OK$,1): IF OK$ <  > "Y" AND
     OK$ <  > "N" THEN 1000
1010  IF OK$ = "N" THEN  POKE 216
     ,0: GOTO 190
1020  PRINT : INPUT "FILENAME: ";
```



SONG WRITER 程式

```
FI$: IF FI$ = "" THEN 1020
1030 D$ =  CHR$ (4): PRINT D$"VER
     IFY"FI$: PRINT D$"OPEN"FI$
1040  PRINT D$"READ"FI$
1050  FOR Z = 1 TO L
1060  INPUT N(Z): INPUT D(Z)
1070  IF N(Z) < = 255 AND D(7) <
     = 255 THEN  NEXT Z: POKF 21
     6,0: PRINT D$"CLOSE":I = Z -
     1: GOTO 190
1080  PRINT : PRINT "INCOMPATIRLF
     FILE!!!": PRINT D$"CLOSE": POKE
     216,0: GOTO 190
1090  PRINT D$"CLOSE": IF  PEEK (
     222) = 5 THEN  POKE 216,0:I =
     Z - 1: GOTO 190
1100  PRINT : PRINT "ERROR#"; PEEK
     (222): PRINT D$"CLOSE": GOTO
     190
1110  REM  HANDLE DOS COMMANDS
1120  ONERR  GOTO 1140
1130 DC$ =  RIGHT$ (A$, LEN (A$) -
     1): PRINT  CHR$ (4);DC$: POKE
     216,0: GOTO 190
1140  PRINT "ERROR#" PEEK (222): PRINT
     CHR$ (4)"CLOSE": POKE 216,0
     : GOTO 190
1150  REM  PRINTER ROUTINE
1160  PRINT : INPUT "PRINTER READ
     Y? ";A$: IF A$ <  > "Y" AND
     A$ <  > "N" THEN 1160
1170  IF A$ = "N" THEN 200
1180  PRINT : INPUT "STARTING NOT
     E TO BE PRINTED --  DEFAULT=
     1: ";ST$: IF ST$ = "" THEN S
     T$ = "1"
1190  PRINT : INPUT "ENDING NOTE
     TO BE PRINTED --    DEFAULT=
     ALL: ";EN$: IF EN$ = "" THEN
     EN$ =  STR$ (I)
1200 ST =  VAL (ST$):EN =  VAL (E
     N$): IF ST < 1 OR ST > I OR
     EN < 1 OR EN > I OR EN < ST THEN
     1180
1210  PRINT : INPUT "NAME OF SONG
     S: ";FI$: IF FI$ = "" THEN 1
     210
1220  PR# 1: PRINT : PRINT FI$: PRINT
     : FOR X = ST TO EN: PRINT "N
     OTE$";X;: HTAB 10: PRINT "NO
     TE=";N(X);: HTAB 19: PRINT "
     DURATION=";D(X): NEXT X
1230  PRINT : PRINT "END OF SONG"
     : PR# 0: GOTO 190
1240  REM  NORMAL PLAY ROUTING
1250  PRINT : INPUT "STARTING NOT
     E (D=1): ";SN$:SN =  VAL (SN
     $): IF SN$ = "" THEN SN = 1
1260  PRINT : INPUT "ENDING NOTE
     (D=LAST):";EN$:EN =  VAL (EN
     $): IF EN$ = "" THEN EN = I
1270  IF SN < 1 OR SN > I OR EN <
     1 OR EN > I THEN 1250
1280  FOR Z = SN TO EN: POKE 768,
     N(Z): POKE 769,D(Z): CALL 77
     0: NEXT Z
1290  GOTO 190
1300  REM  INSERT ROUTINE
1310  POKE 216,0: PRINT : INPUT "
     INSERT BEFORE WHAT NOTE? ";I
     B: IF IB < 1 OR IB > I THEN
     1310
1320  PRINT : INPUT "HOW MANY NOT
     ES TO INSERT? ";HM: IF HM >
     L - I OR HM < 1 THEN 1320
1330  FOR Z = IB TO IB + HM - 1
1340  PRINT : INVERSE : PRINT "NO
     TE#"Z: NORMAL : INPUT "NOTA,
     DURATION: ";NN(Z),ND(Z): IF
     NN(Z) < 0 OR NN(Z) > 255 OR
     ND(Z) < 0 OR ND(Z) > 255 THEN
     1340
1350  POKE 768,NN(Z): POKE 769,ND
     (Z): CALL 770
1360  NEXT Z
1370  FOR Z = I TO IB STEP  - 1:N
     (Z + HM) = N(Z):D(Z + HM) =
     D(Z): NEXT Z
1380  FOR Z = IB TO IB + HM - 1:N
     (Z) = NN(Z):D(Z) = ND(Z): NEXT
     Z
1390 I = I + HM
1400  GOTO 190
1410  REM  DELETE ROUTINE
1420  PRINT : INPUT "DELETE FROM
     NOTE (D=LAST): ";DF$: IF DF$
     = "" THEN I = I - 1: IF I =
     - 1 THEN I = 0: GOTO 190
1430  IF DF$ = "" THEN 190
1440  PRINT : INPUT "TO NOTE: ";D
     T$:DF =  VAL (DF$):DT =  VAL
     (DT$): IF DT < 1 OR DT > I OR
     DF < 1 OR DF > I OR DF > DT THEN
     1420
1450  FOR Z = DT + 1 TO I:N(Z - (
     DT - DF + 1)) = N(Z):D(Z - (
     DT - DF + 1)) = D(Z): NEXT Z
1460 I = I - (DT - DF + 1): GOTO
     190
```

DOS的BOOT碟原理

DOS 3.3專欄

甘貴祥

DOS 是以滙編語言寫成的程式，當使用 BOOT 磁碟時，它便放入 RAM 記憶體中。BOOT MASTER DISK 時，DOS 映象依所使用機器的大小不同而放入其 RAM 的末段位址上。如此一來，DOS 就必須使 BASIC 了解到它可用到的RAM 少於系統原有確實的數量。例如，在 48 K APPLE II 之中作使DOS 時，BASIC 只有 38 K RAM 可供利用。DOS 放入記憶體後，以調整 HIMEM 的方法防止 BASIC 使用到 DOS 所佔的記憶體。BOOT SLAVE DISK，DOS 放入起始設定磁時（INITIALIZED）它所在的位置，若 SLAVE DISK 建於16 K APPLE 上，DOS 便放入第 6 至16 K RAM範圍中。如此一來，APPLE 實際上只剩 6 K RAM 可供應用。若 SLAVE DISK建於 48 K APPLE上，它便無法在小於48K系統上 BOOT ，因爲 SLAVE DISK 上 DOS所佔的RAM 位置在較小系統上不存在。

圖（1）顯示 48 K APPLE II DOS 的記憶位置安排情形。其中含有四個主要區間，爲首的 1.75 K 爲檔案緩衝器（FILE BUFFERS）。由於自然値（DEFAULT）定爲 MAXFILES 3 ，此處含有3 組檔案緩衝器。每個緩衝器對應一個 OPEN FILE ，佔有 595 個 BYTES。DOS 利用檔案緩衝器處理 LOAD 及 SAVE FILE 等工作。若將 MAXFILES 數目改掉，則檔案緩衝器所佔的空間發生變化，影响 HIMEM指向器，使其根據緩衝器空間之變化往上往下移動。

檔案緩衝器上方 3.5 K位置存 DOS ROUTINES 可執行的 DOS 機器語言碼起始於此。ROUTINES 功能包括起始設定 DOS ，對 BASIC 介面，解釋命令（COMMANDS ）及處理檔案緩衝器事務。所有磁碟的 FUNCTION 都經由呼叫檔案處理器而傳出執行。

佔 2.8 K的檔案處理器集合了許多負責執行各種存取磁碟所須的功能。這些功能包括 OPEN，CLOSE，READ，WRITE，POSITION，DELETE，CATALOG，LOCK，UNLOCK，RENAME，INIT 及 VERIFY。檔案處理器雖是在DOS 之中的副程式，使用者的滙編語言程式仍可透過介面呼叫它。此介面概括爲一群在 RAM 第三頁的向量指向器（VECTORS）DOS最後的 2.5 K位置是R／W，T／S（讀／寫，軌道／段落）（RWTS）包製組。RWTS 是檔案處理器的次級規約，實際上它只是檔案處理器所呼叫的副程式。檔案處理器處理檔案，而 RWTS 處理磁碟上軌道及（TRACK)段落（SECTOR）的問題。呼叫 RWTS 的典型假子如將 IT軌道第 0 段落中之資料讀出，或將記憶體中 256個BYTES資料寫入第 5 軌道第E段落中。DOS 也具有外部介面好讓使用者的滙編語言程式可得用得到 RWTS。

記憶體第三頁的DOS向量指向器

有大約10K高位址 RAM 被 DOS 佔用以外，在低位址記憶體的第 3 頁處（$300至$3FF），DOS 亦保有空間以儲存一群向量指向器，經由某固記位置（如：$3D0）可利用此向量指向器進入 DOS 子常式集的特定點。因 DOS 放入記憶體的位置隨著機器大小及使用的磁碟而不同。在 DOS 之中可由外界呼叫子程式其位址並不固定。若將子程式位址存於固定位置的向量指向器，此困難便迎刃而解，至於不同版本的 DOS ，其子程式位置亦可能不同。此時第 3 頁的向量指向器在定位這些子程式方面亦大有貢獻。DOS 早期版本在 BOOT 過程存入BOOT 1程式時使用到 $200 至$3CF 位置，但在 DOS 3.3 中，此位置作資料緩衝器及磁碟碼轉換表。也許，這項更改的目的在於可供應較多的記憶體給予第一階段BOOT截入程式使用，向量指向器表本身則自$3D0 開始。

DOS向量指向器$3D0－$3FF說明

地址	使用說明
3D0	存 JMP 指令，將程式跳越至 DOS 的 WARM－START ROUTINE。這子常式讓使用者重入 DOS，但並不取消在執行的 BASIC 程式，也不重置 MAXFILES 或其他環繞DOS 的變數。
3D3	存 JMP 指令，將程式跳越至 DOS 的 COLD－START ROUTINE，以重新起始設定 DOS。與 REBOOT DOS 的作用一樣，它清除了正在執行的 BASIC 檔案並重置 HIMEM。
3D6	存入 JMP 指令，將程式跳越至 DOS 的檔案處理器，讓使用者的組合語言程式呼叫它。
3D9	存 JMP 指令，將程式跳越至 DOS 的 RWTS 子常式，讓使用者的滙編語言程式可呼叫它。
3DC	呼叫一短的子程式以存放檔案處理器所須的輸入參數表，讓使用者在呼叫檔案處理器之前可先設定適當的輸入參數。
3E3	呼叫一短子程式以存放 RWTS 所須的輸入參數表，讓使用者在呼叫 RWTS 之前可設定適當的輸入參數。
3EA	使 JMP 指令，將程式跳越至一子程式，以便重新連接 DOS 截斷器（INTERCEPT）到鍵盤及 DATA STREAM上。
3EF	存 JMP 指令，將程式跳越至處理機器語言指令“BRK” 的子常式。內含顯示暫存器內容的子程式

（在監督程式 ROM 之中）的位址。本向量指向器僅由自動啓動 ROM（AUTOSTART ROM）給定。

3F2 存有供自動啓動 ROM 使用的 RESET 子常式的位址（LOW BYTE ／HIGH BYTE），存著 DOS 重新啓動（RESTART）的位址，但使用者若欲自行處理 RESET 動作時，可更改它。

3F4 POWER—ON BYTE，內存與 RESET 位址互補的值 $A5，用以判別是機關動作或是 RE—SET 按鍵動作。開機時，自動啓動 ROM 及執行在 $3F2 的功能（因尚未起始設定）而執行 BOOT 磁碟過程。爲了避免使用者於更改 $3F2 內容以處理自己的RESET時，發生誤認爲開機事件，使用者應將 $3F2內的值與 $ A5 作 EOR（EXCLUSIVE OR）運算再存回 $3F4 中。

3F5 將程式跳越至一機器語言子常式。在執行 APPLESOFT，當使用到“&”特徵符號（FEATURE）時呼叫此常式。

3F8 將程式跳越至一機器子常程，當“CONTROL—Y”鍵入時呼叫此常式。

3FB 將程式跳越至一機器語言常式，當不可罩（NO—NMASKABLE）中斷發生時呼叫此常式。

3FE 存有一常式的位址（LOW—BYTE／HIGH—BYTE），當可罩（MASKABLE）中斷發生時呼叫此處的常式。

BOOT碟到記憶體的過程

APPLE開機時，記憶體中並無任何程式。 BOOT 磁碟的獲取 DOS. BOOT 就是將 DOS 的靴式載入 RAM 的過程。 BOOT 過程包括一系列步驟，把 DOS 程式的片段自小至大依次序地載入記憶體中，直到所有程式完全進入記憶體中並可用來執行爲止。 BOOT 在此DOS 中分爲四個階段。這四個階段在磁碟及記憶體上的對應位置可由圖（2）得之。以下詳述 BOOT 過程。

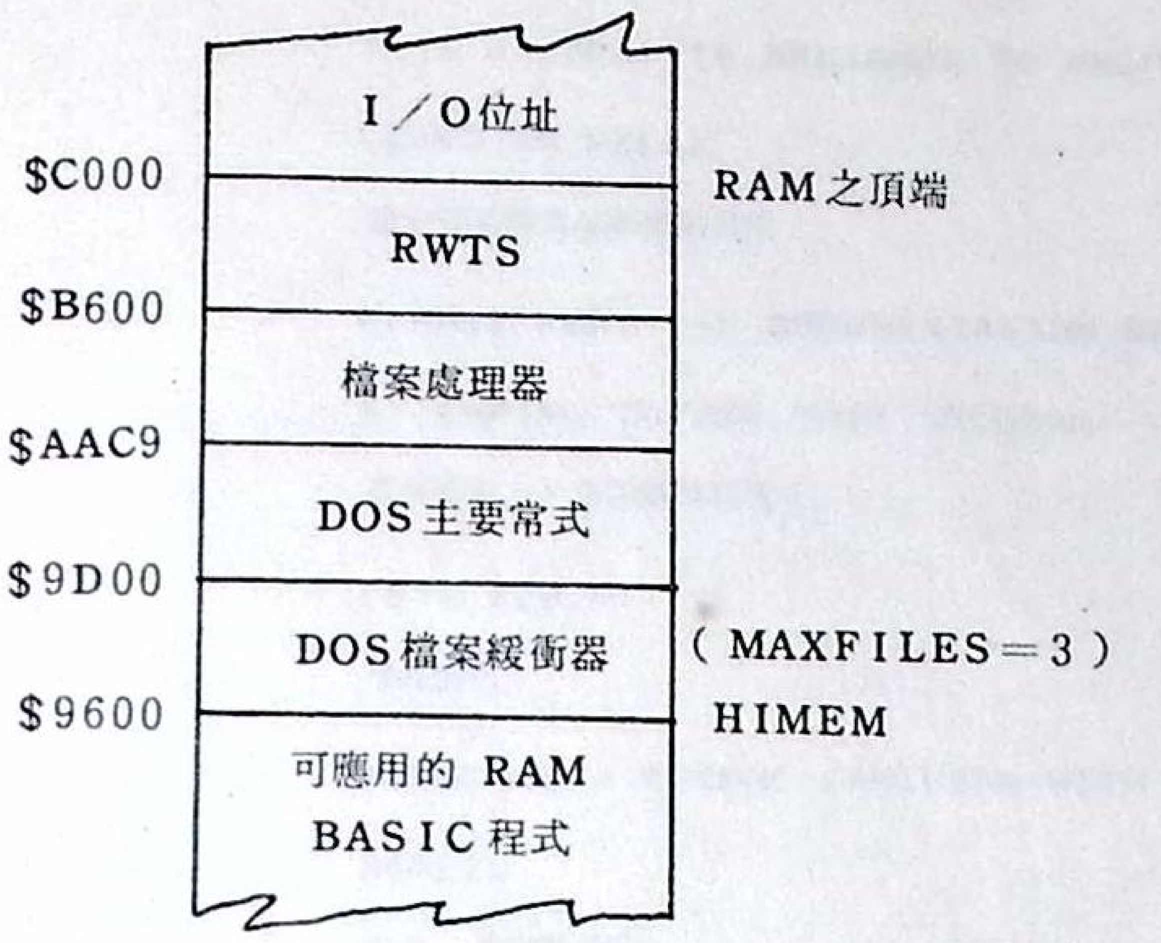
第一階段 BOOT（稱爲 BOOT O）執行儲存於磁碟控制卡上 ROM 中的程式。使用者鍵入 PR# 6,C600 G 或 6 個P（↑表 CONTROL），CPU 控制權便交由在第六接綫槽上磁碟控制卡的ROM。此ROM 中含有 256 個 BYTE 的機器語言程式。當執行時，它將磁碟探臂拉回第零軌道上的重校準探臂位置（可聽到咔啦的聲音）。然後將第零軌道第零段落的內容自RAM 的 $800 處讀入RAM（早期DOS 版本自$ 300處讀入）。讀完後第一階段 BOOT 便令程式跳越至$ 800 位置進行第二階段 BOOT的工作（稱爲 BOOT 1）。

BOOT 1 也有256 個 BYTE，它利用了部份BOOT 0 ROM 中子程式將第零軌道上另外 9 個段落（第 1 段至第 9 段）內容讀入 RAM 中。這些段落內含次階段 BOOT 過程（稱爲 BOOT 2）。根據所使用磁碟種類的不同，BOOT 2 載入記憶體中不同的位置。若使用副磁碟， B—OOT 2 載入DOS 所在記憶體最後 9 頁（每頁 256 個 BYTE）位置上，此處即副磁碟最初被起始設定的地方，也就是，若副磁碟原建於 32 K DOS，BOOT 2就會載入 $7700 至 $8000 RAM中，若使用主磁碟，BOOT 2 載入 $3700至$ 4000 位址 RAM中與副磁碟在16K系統情況下相同。於載入 BOOT 2 的記憶體下方，再度載入 B—OOT 1，（主磁碟情形下是在$ 3600），以便當起始設定一新磁碟時可直接將BOOT 1，複製到磁碟的第零軌道第零段落上，當 BOOT 1 執行完且載入 BOOT 2 之後，程式便跳越至 BOOT 2 的進行下階段 BOOT工作。

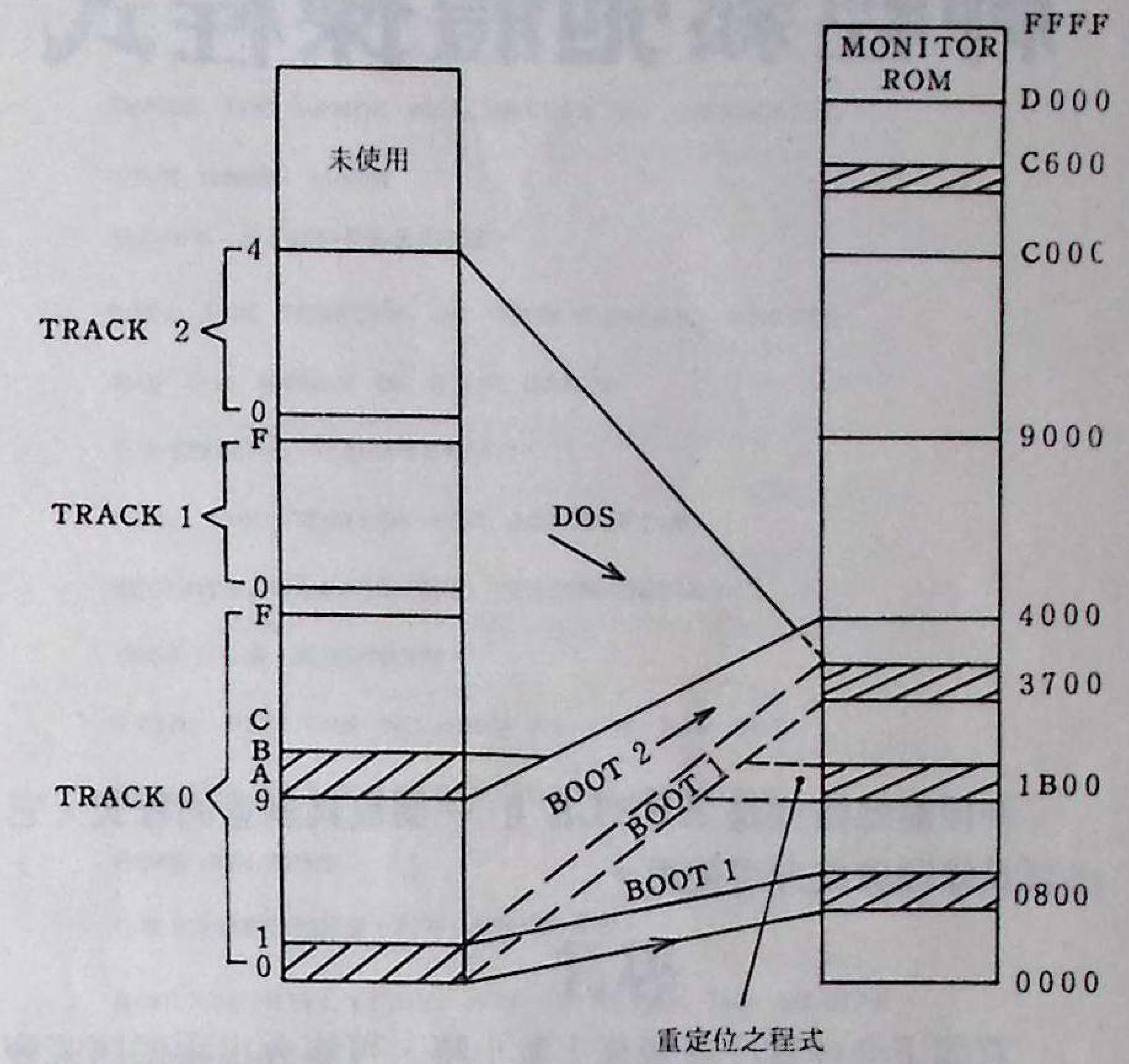
BOOT 2 包括載入程式（LOADER）的“主程式”及 RWTS 子程式。到此爲止磁碟探臂因僅在第零軌道上讀取而未移動過。此後探臂須移至其他軌道上以取得更多段落資料的載入記憶體。處理複雜的磁碟存取工作， BOOT 2 載入程式利用 RWTS 子程式專責控制磁碟探臂的移動並且將 BOOT 最後一部份（也就是 DOS 本身）所在的段落內容載入記憶體中。BOOT 2 將探臂定於磁碟第二軌道第四段落上，並將其內容讀入記憶體中 BOOT 1 所在位置的下方（使用主磁碟時，位址爲 $3500 ）。然後 BOOT 2 連續讀入 26 個段落資料（除了最後一段外，每段有 256個 BYTE）。使用主磁碟時，最後並將第零軌道第A段內容自$ 1B00讀入。這 27 個段落內容就是DOS 主常式群以及檔案處理器。經過這些載入過程後所有的DOS 均已在記憶體之中了，此時副磁碟的 BOOT 過程已大功告成，程式並跳越至 DOS COLDSTART 的位址去。在使用主磁碟情形下，此時僅16K APPLE Ⅱ 才能使用在記憶體中D—OS 的映象，若系統大於 16K， DOS 必須再移到最高的記憶位置上去。因此程式先跳越到 $1B03重定位程式位址上。重定位程式長 512個 BYTE，它已事先存於DOS 映像的最低兩頁中（使用副磁碟時，這兩頁塡入“零”）。重定位程式有系統地做儲存及載入的動作於較高頁RAM，以決定機器大小直到找到最高頁爲止。之後它便將 DOS 自$ 1D00 往上移，移到規定的位置去（48 K系統中爲$9D00），並利用表格修正程式中機器碼使程式在新位置上能夠正常工作。然後重定位程式便跳越至最高位置 DOS 上並將低位置DOS 捨棄掉。

COLDSTART起始設定 DOS，爲檔案處器準備空間

DOS的BOOT碟原理



圖（1） DOS之結構



，預定 HIMEM，建立第三頁向量指向器表並執行 HELLO 程式。DOS BOOT 過程到此全部完畢。

早期 DOS 版是的 BOOT 過程稍嫌繁複，它們的BOOT 1 載入＄300，及 BOOT 2載入＄3600與 DOS 3.3 相同，但它們的 27 個段落並不在全部載入記憶體。副磁碟情況下，只有 25 個段落載入，且若所使用的磁碟爲13個段落式的（每一軌道），其DOS 映像的終止位置依副磁碟或主磁碟不同，分別在第二軌道第 8 或第A段落。並且其 BOOT 1，在磁碟中的分塊（NIBBILIZATION）型式與其他段落不同，單純地將它載入記憶體中並不能用來執行。□

各期電腦時代程式磁碟歡迎補購

第一期程式磁碟

	附程式之文章	在磁碟索引表中之相對檔案名稱
1	突破傳統的鍵盤	DVORAK KEYBOARD
2	捉球遊戲	BEND OF ROUNDING
3	勇闖電腦室	GAME OF SABOTAGE
4	迷宮救美	GAME OF MAZING
5	將 APPLESOFT 變爲 FORTRAN	MAZE SHAPES APPLESOFT TO FORTRAN
6	上、下、左、右移動文字畫面	PEELING THE TEXT SCREEN PEELING LISTING 5
7	測試工具一 BOOLEAN 程式	BOOLEAN TUTORIAL BOOLEAN TUTORIAL—MAIN PROGRAM
8	騰空更多磁碟空間一搬移HELLO	GREETING TRANSFER GREETING
9	重現刪除之檔案	RETRIEVE THE DELETED PROGRAM
10	電腦時代月曆	APPLE CALENDAR APPLE CALENDAR & SUM SUM 1 CAL.FILE—1983 CAL.FILE—1984 CAL.FILE—1985 CAL.FILE—1986
11	COMPUTEXT 文字處理	COMPUTEXT
12	時日相距計算程式	DYNAMIC CALENDAR
13	增強英文字彙能力	GAME OF SPELLING TEST
14	幼童數學四則運算練習	GAME OF MATHEMATIC
15	過四關	GAME OF CONNECT FOUR

第二期程式磁碟

	附有程式之文章	在磁碟索引表中之相對檔案名稱
1	貨櫃處理	CONTAINER HANDLING
2	沙蟹幻方局	POKER LOADER POKER SHOWHAND
3	巧破密令	GAME OF CRYPTO
4	美術字繪劃器	RETRIEVE FONTASTIC SCROLL!! SUPERFONTASTIC LEAD CHAR 1
5	小階字母處理	LOWER CASE CHIP CONVERTER
6	程式轉載器	SAFELY HANDLING PROGRAM
7	彩色電視／監視器色彩測試	CRT COLOR DEMONSTRATION TEST
8	保留遊戲積分記錄	HIGH SCORE KEEPER
9	數學套件程式	MATHS TOOL KIT 1
10	磁碟修護工具	Z BPHTCH
11	加送電腦時代月曆圖像一個	PATCH 2 LOADSUM 2 SUM 2

請前往COMPUHOUSE補購

地址：深水埗南昌街白田購物中心（近南昌戲院）二樓197室　　電話：3-7776383

神經病醫生

神經鬆弛體操程式

神經鬆弛體操是 APPLE II 一個很具創意的程式，它必須具備磁碟機方可使用。

導言

若閣下全神貫注在螢幕上數小時，可能會出現頭痛或疲倦現象，有時這種現象會維持數小時甚至數天之久。而且過份精神緊張，視覺疲勞，不正確的坐姿所引起的不舒適就如罪犯的煩惱一樣纏繞著，例如其中一個或多個部位的肌肉拉緊。通常這些情形在一個熱水浴之後便可回復正常，但若這個方法失效時，閣下可以嘗試以電腦來幫助鬆弛，這個程式的內容可幫助閣下鬆弛任何肌肉出現的疲勞現象。而且你可以不必花費，安坐家中的電腦前來完成。

鬆弛原理及實習

本程式是筆者參考兩位美國心理學家 THOMAS C. SIMEK及 RICHARD M. O'BRIEN 所合著名為 TOTAL GOLF 內所討論的內容改編而成。在 TOTAL GOLF 一書內，他們發表了一個系統性強而有趣的途徑來自行達到鬆弛途徑。

其原理是十分簡單的，因為當緊張之時，肌肉會開始疼痛，它是不會自己放鬆，除非你懂得如何去鬆弛它們，如此一來會令人們以烟、酒、麻醉藥品或以物理治療方式如對疼痛位置熱療等來解決，而另一方法是你自己以經常性的高敏銳能力以第一時間提醒自己某部份過於緊張而自予性發以鬆弛，這種方式是基於預防形式來達成。原則上本程式是希望各位讀者將被訓練成對這些身體上的暗示訊息以第一時間來鬆弛。而訓練的過程是利用一些螢幕及音响訊息。通過程式的運行，來鬆弛身體上各部份肌肉，而且基於電腦訊號對這些部位拉緊及鬆弛可望於短期內達到隨心所欲的境界。

如何開始鬆弛

（1）確定無任何發光體令螢幕反光。

（2）置身於螢幕前，頸項肌肉不應緊張而可舒適看到它。

（3）將螢幕的光暗調校至柔和程度，刺眼的白色映像應調校至灰白的狀態。

（4）閣下應坐於有扶手的椅子前，以舒適為原則，雙脚踏在地板上。

（5）離螢幕約 2 ½ 至 3 呎左右。

注意：在進行鬆弛之前，不可配戴隱形眼鏡，事先除下它。第二，若閣下有關節炎的話，請諮詢醫生這種動作是否適合。第三，若肌肉因意外如破損等之時，不可使用本程式。

鬆弛肌肉範疇

本程式將肌肉組織區分為以下各組以進行訓練：

（1）手、二頭肌（即俗稱老鼠仔）及三頭肌。

（2）額部、鼻部及口部。

（3）頸項、肩部及胃部。

（4）大腿、小腿及脚部。

在本程式運行時，螢幕上會出現身體上各部份肌肉位置的英文代碼。並且會出現一些文字說明，此外還加上音頻訊號來作為輔助，以提高效率。

在程式運行時，MENU 上會出現

R —REVIEW

E —ENTIRE PROGRAM

D —DEEP BREATHING TRAINER

"R"是供閣下在全個程式運用純熟後，（即 ENTIRE PROGRAM）所簡化的動作過程。

"D"是一個稱為深呼吸訓練法，利用它可以培養出深切而有節奏的呼吸節拍，使用時只須在吸氣時按入歡樂棒的按鈕"0"，呼氣時按入"1"鍵。

一些備忘錄

當使用主程式 RELAX—MAIN PROGRAM 時，在人體圖形下的一條粗橫綫是計時器，它隨著時間增長而變動。

副程式 CREAT RELAX FILES 是用以產生兩個重要機械程式，包括圖形表及運行程式。它們的 FILE 名稱分別為 RELAX ROOTS及 RELAX TEXT，因此當主程式運行時確定這些 FILES 在同一面磁碟上。希望閣下看完本文後能 RELAX 一下！

在運行「電腦時代程式磁碟」時，先 RUN CREAT RELAX FILES，然後再 RUN RELAX MAIN PROGRAM 即可。

中英對照指示

本程式執行時，主要以文字告訴用者，其中有些英文生字並不常用，爲使英文程度較低的讀者也能分享此神經鬆弛體操的樂趣，我們特別將畫面上的文字全部譯爲中文，供讀者參考。

THIS PROGRAM IS DESIGNED TO HELP YOU LEARN TO RELAX

這程式是教你如何鬆弛肌肉

PLEASE READ THE DOCUMENTATION BEFORE ATTEMPING TO USE THIS PROGRAM

在使用前，請先看清楚說明。

LETS BEGIN

現在開始。

FIRST LETS BECOME FAMILIAR WITH THE SCREEN

首先，請留意螢幕。

ONE THE RIGHT SIDE OF THE SEREEN IS A FIGURE

螢幕右面是一個人體圖形。

TEXT WILL APPEAR TO THE LEFT OF THE FIGURE

螢幕左面用來顯示文字。

AT THE BOTTOM OF THE SCREEN,THERE IS A TIMER BAR

螢幕底邊的計時綫用作計時器。

THE FIRST FEW TIMES THAT YOU USE THIS PROGRAM,YOU WILL NEED TO PAY CLOSE ATTENTION TO THE TEXT.AFTER A WHILE,YOU WILL LEARN THE RELAXATION INSTRUCTIONS AND YOU WILL ONLY NEED TO LOOK AT FIGURE AND LISTEN TO THE AUDITORY CUES TO DO THE EXERCISES

初學這些運動，你先要留心螢幕的指示，當你熟習後，只需要看右面的圖形和電腦發出的電子音樂。

TO PREPARE,IMAGINE A TRANQUIL SCENE

準備運動時，心中要幻想着一個寧靜的環境。

CONCENTRATE ON THIS SCENE AND ON YOUR BREATHING SO IT IS DEEP AND RHYTHMIC

集中精神，使你的呼吸更深，節奏更穩定。

CLOSE YOUR EYES AND THEN OPEN THEM AGAIN AT THE TONE

閉上眼睛，直至聽到音樂。

THE FIRST MUSCLE GROUP YOU WILL RELAX IS THE HANDS,BICEPS AND TRICEPS GROUP

你需要鬆弛的第一組肌肉是手部，二頭肌和三頭肌。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THIS MUSCLE GROUP

這聲音是指示你鬆弛這個肌肉。

BEGIN WITH THE HANDS,THIS TONE,SIGNALS YOU TO FOCUS ON YOUR,HANDS

先由手部開始，這音樂聲是提示你做手部運動。

TENSE THE HANDS AND,WRISTS BY SQUEEZI IG YOUR HANDS HARD

緊握雙手，使手部和手腕肌肉收緊。

FEEL THE TENSION IN YOUR FINGER, WRISTS AND THE BACKS OF YOUR HANDS

你會感覺到手指，手腕和手背都受力。

FEEL THE TENSION FOR ABOUT FIVE SECONDS,RELEASE THE TENION SLOWLY

保持約五秒鐘，然後緩緩放鬆。

ENJOY FEELING RELAXED AS THE BAR GETS SMALLER.LET YOURSELF BECOME MORE AND MORE RELAXED

你會享受到鬆弛的好處，同時，計時綫亦變動。

NOW YOU WILL LEARN HOW TO RELAX THE BICEPS

現在輪到鬆弛二頭肌。

THIS TONE WILL SIGNAL YOU TO RELAX THE BICEPS

這音樂聲會指示你鬆弛二頭肌。

THIS IS DONE BY SQUEEZING THE BICEPS,THE WAY YOU DO WHEN SOMEONE SAYS (MAKE A MUSCLE).DO NOT MAKE A FIST WITH YOUR HANDS

先收緊二頭肌，這動作就是我們俗稱的「鼓起老鼠仔」。但手部不要緊握拳頭。

BE SURE TO REMEMBER TO DO THIS WITH BOTH ARMS

別忘記要一對手臂的二頭肌也一齊收緊。

NOW LET THE BICEPS RELAX,PUSH "SPACE" AND FOCUS ON THE BAR....

現在放鬆二頭肌，按" SPACE BAR"。

NOW YOU WILL RELAX THE TRICEPS AND TOPS OF THE FOREARMS

現在輪到鬆弛三頭肌和前臂上部。

THIS TONE WILL SIGNAL YOU TO RELAX THE TRICEPS AND UPPER FOREARM

這聲响是指示你鬆弛三頭肌和前臂上部。

TO FEEL WHAT IT IS LIKE,TO TENSE THESE MUSCLES PRESS YOUR FOREARMS DOWNWAND INTO THE ARMS OF A CHAIR,FEEL THE TENSION FOR ABOUT FIVE SECONDS

你可以用前臂力壓椅的手墊五秒鐘。

NOW RELAX...LET YOURSELF RELEX EVEN MORE AS THE BAR GETS SMALLER AND SMALLER.PLEASE PRESS THE SPACE BAR

現在慢慢放鬆，直至「計時綫」漸漸消失，按" SPACE BAR"。

NOW YOU WILL LEARN HOW TO RELAX THE FORAHED,NOSE AND MOUTH

現在大家學習如何鬆弛前額、鼻和口部。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THIS MUSCLES GROUP

這聲响是提示你準備鬆弛上述肌肉。

THIS TONE WILL SIGNAL YOU TO RELAX THE FOREHAND

這聲响是提示你鬆弛前額。

TENSE THE FOREHEAD BY FROWING HARD,HOLD THE FROWN FIVE SECONDS

皺起前額肌肉五秒。

RELAX THE FROWN AND ENJOY THE PLEASANT SENSATION

放鬆前額。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THE NOSE

這聲响是指示你準備鬆弛鼻部。

FIRST TENSE THE MIDDLE OF YOUR FACE BY SCRUNCHING UP THE NASAL AREA

首先收緊鼻部令面部中央受到壓力。

TRY TO KEEP THE FOREHEAD SMOOTH,TENSE THIS AREA FOR ABOUT FIVE SECONDS

保持這部位收緊五秒鐘，而放鬆前額。

NOW BEGIN TO RELAX THIS AREA ...

現在開始鬆弛面部肌肉。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THE MOUTH AREA

這聲响是叫你準備鬆弛咀部肌肉。

TENSE THE MONTH BY PRESSING THE LIPS TOGETHER HARD

緊閉上下唇，使咀部肌肉收緊。

HOLD THE TENSION FOR ABOUT FIVE SECONDS,THEN LET YOURSELF

持續收緊五秒鐘。

NOW YOU WILL LEARN HOW TO RELAX THE NECK,SHOULDERS AND STOMACH

現在大家來學習如何鬆弛頸部、肩膊和腹部。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THE MUSCLE GROUP

這聲响是通知你準備鬆弛上述肌肉。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THE NECK MUSCLES

這聲响是指示你準備鬆弛頸部肌肉。

TENSE THE NECK BY PLACING A HAND AGAINST YOUR FOREHEAD AND PRESSING YOUR HEAD INTO YOUR HAND.DO THIS FOR FIVE SECONDS,THEN.....

將前臂放在額前，頭向前壓，使頸部肌肉受力，持續五秒鐘，然後...

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THE SHOULDERS

這聲响是告訴你下一步是鬆弛肩頭。

TENSE THE SHOULDERS BY PUSHING THEM INTO YOUR CHAIR

將肩膊壓向椅背，使它們受壓力。

HOLD THE TENSION FOR ABOUT FIVE SECONDS,THEN LET YOURSELF

持續約五秒鐘後放鬆。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THE STOMACH

這聲响是告訴你下一步是鬆弛腹部。

TENSE THE STOMACH BY SUCKING IN THE STOMACH HARD

盡力提起腹部，使腹部受力收縮。

HOLD THE TENSION FOR ABOUT FIVE SECONDS,THEN LET YOURSELF

持續收縮約五秒鐘鬆弛。

NOW YOU WILL LEARN HOW TO RELAX THE THIGHS,CALVES AND FEET

現在，大家學習如何鬆弛大腿、小腿和雙脚。

THIS TONE WILL SIGNAL YOUR TO PREPARE TO RELAX THIS MUSCLE GROUP

這聲响是提示你準備鬆弛上述肌肉。

THIS TONE WILL SIGNAL YOU TO PSEPARE TO RELAX THE THIGH MUSCLES

這聲响是提示你準備鬆弛大腿肌肉。

TENSE THE THIGHS BY STRETCHING YOUR LEGS OUT IN FRONT OF YOUR BODY AS FAE AS YOU CAN.DO THIS FOR FIVE SECONDS THEN.....

盡你所能將雙腿伸前，持續五秒鐘，使大腿受力，然後……

THIS TONE WILL SIGNAL YOU THE CALF MUSCLES

這聲响是告訴你下一步是鬆弛小腿肌肉。

TENSE THE CALVES BY RESTING YOUR HEELS ON THE FLOOR WITH YOUR LEGS OUT IN FRONT OF YOU.NOW DRAW YOUR FEET UPTOWARDS YOUR FACE.....FEEL THE TENSION AND THEN LET THESE MUSCLES

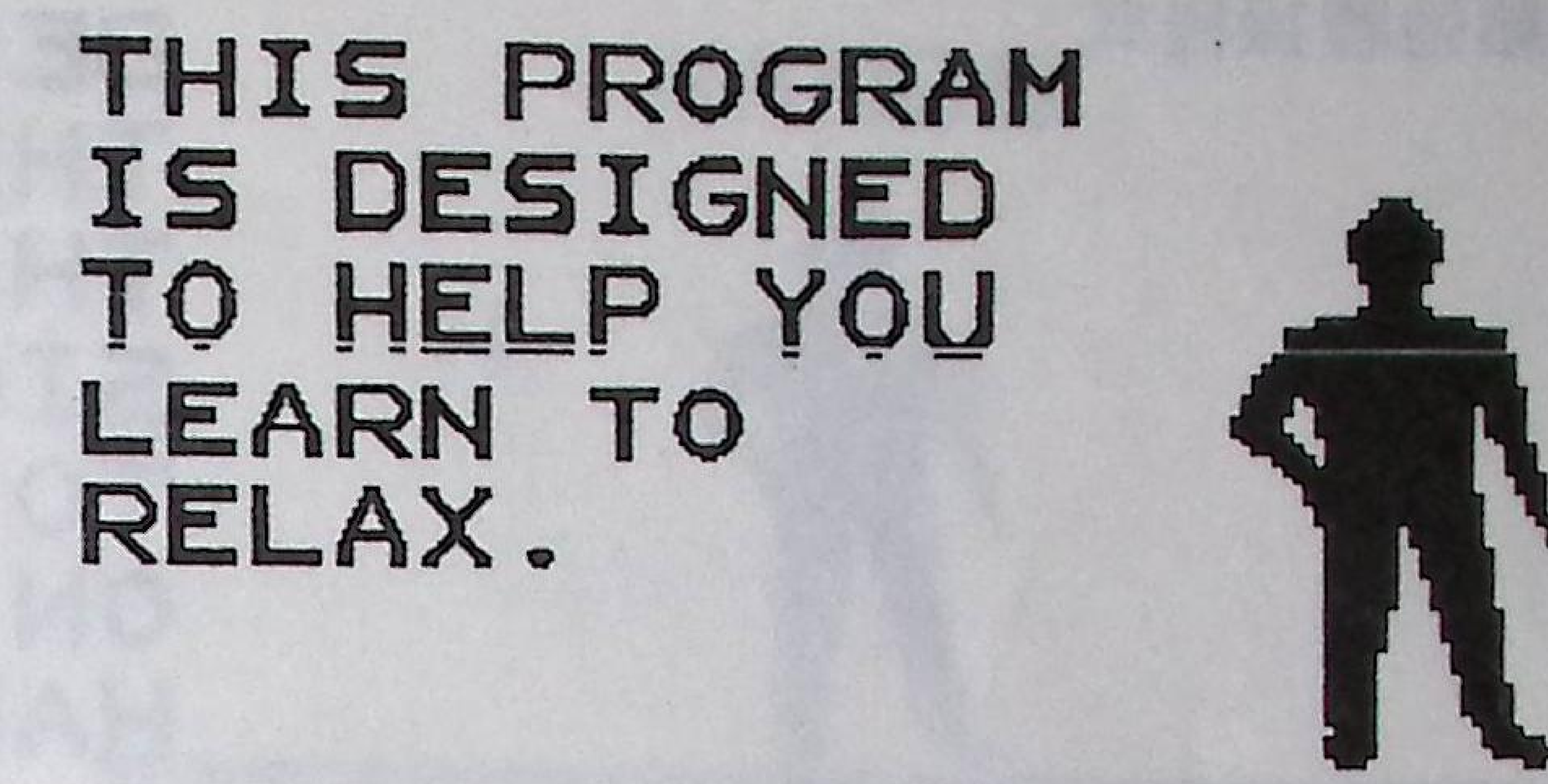
將脚後跟放在地上，脚向前伸，將雙脚向上提起，使小腿肌肉受力。

THIS TONE WILL SIGNAL YOU TO PREPARE TO RELAX THE FOOT MUSCLES

這聲响是告訴你下一步是鬆弛足部肌肉。

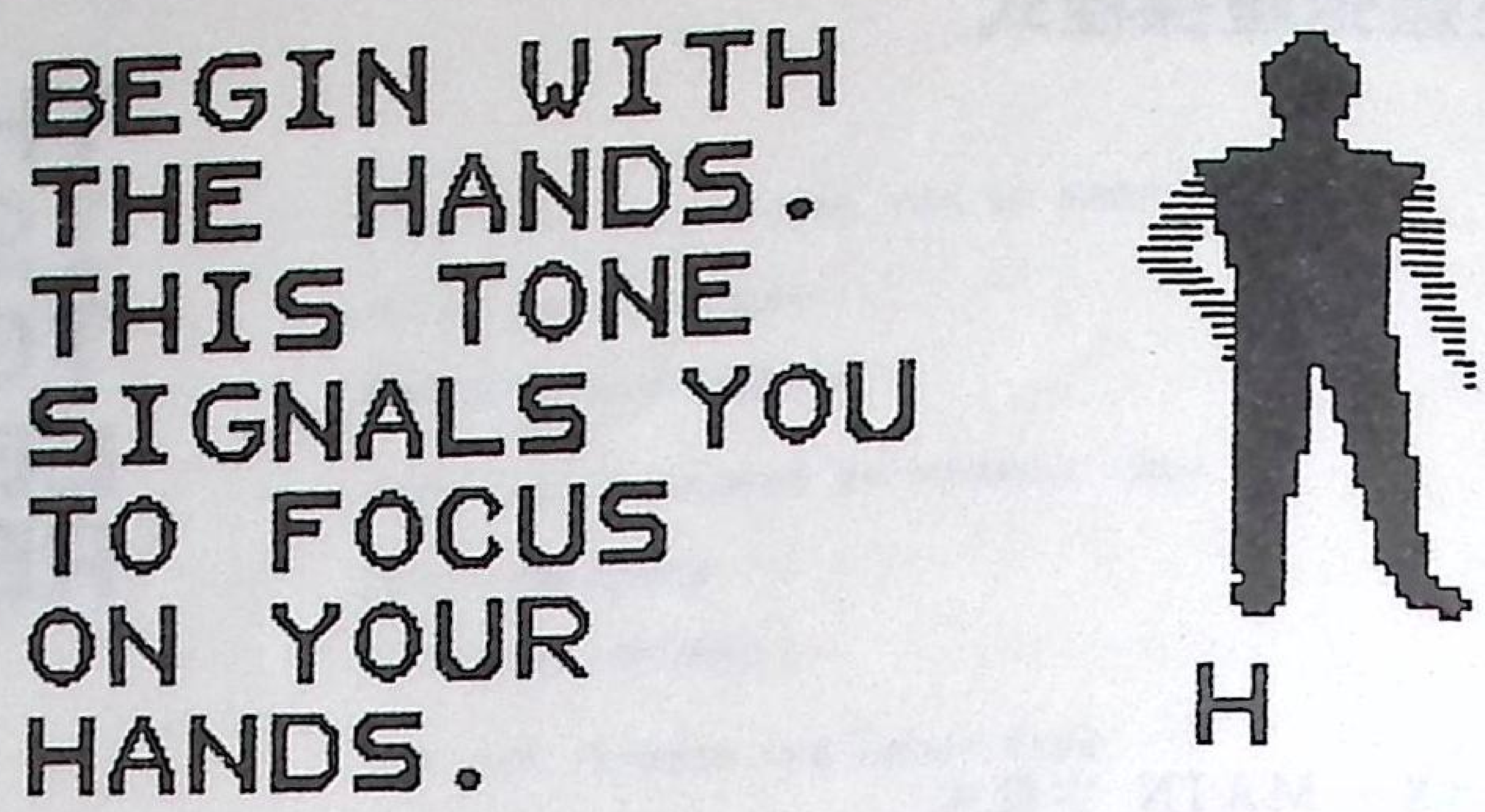
TENSE THE FEET BY CURLING YOUR TOES INWARD AND AQUEEZING THEM HARD.TENSE THE FEET THIS WAY FOR ABOUT FIVE SECONDS,THEN....

將脚趾緊貼並向內屈曲，保持這動作約五秒鐘，然後……。

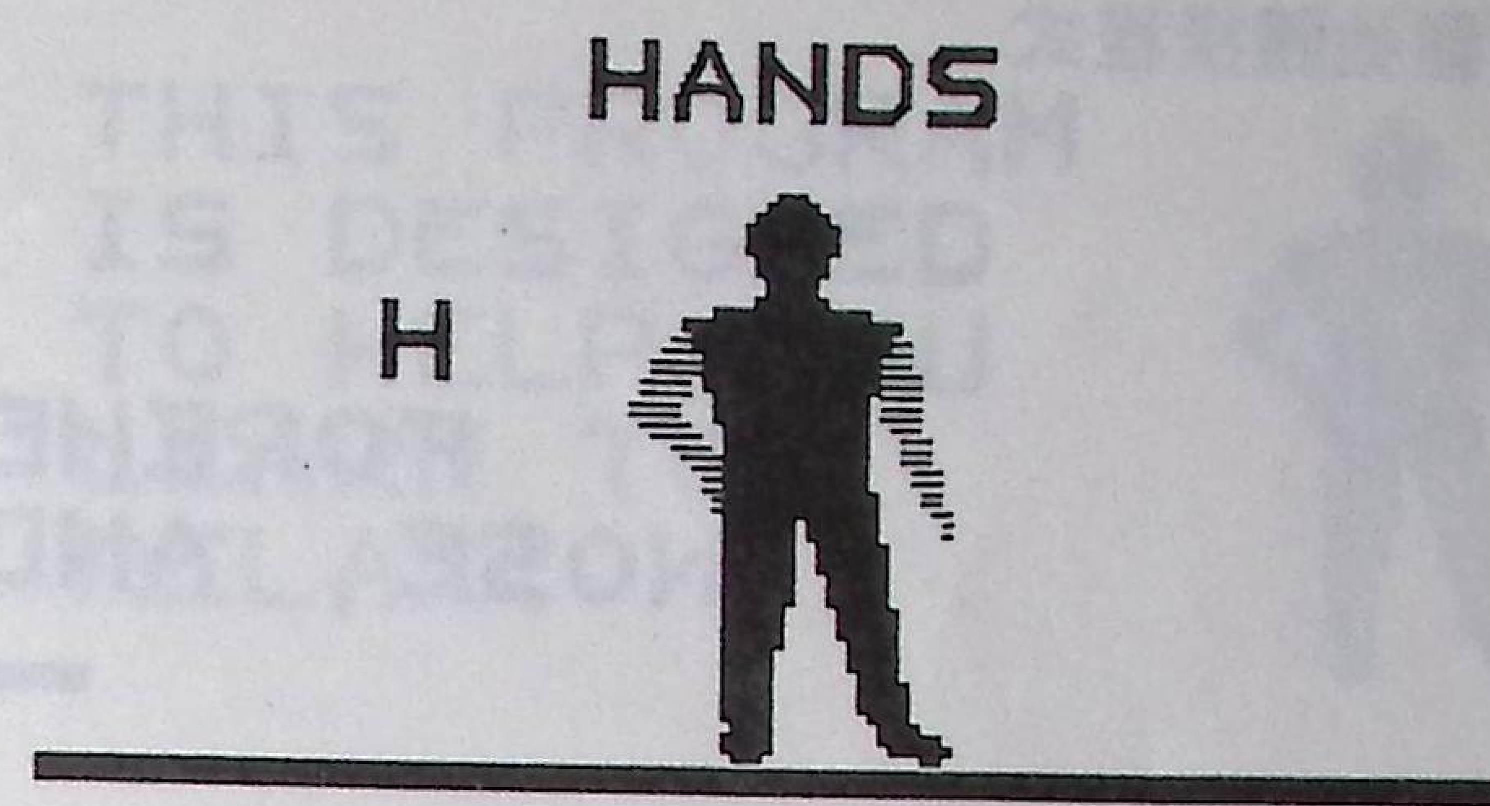


RELAX—MAIN 主程式

```
100  HIMEM: 16384:D$ =  CHR$ (4):
      PRINT D$"BLOAD RELAX TEXT":
      PRINT D$"BLOAD RELAX ROUTS"
     : POKE 1013,76: POKE 1014,0:
      POKE 1015,3: POKE 232,0: POKE
     233,96: SCALE= 1: ROT= 0
110  FOR X = 1 TO 9: READ TN(X): NEXT
     : DATA  185,90,158,69,190,34
     ,20,120,30
120 KBD = 49152:CLK = 49168:B0 =
     49249:B1 = 49250
130  &  RESTORE  GOTO 1800: GOSUB
     290: POKE CLK,0
140  GET A$: IF A$ > "Z" THEN A$ =
      CHR$ ( ASC (A$) - 32)
150  IF  ASC (A$) = 27 THEN  TEXT
     : HOME : END
160  IF A$ = "D" THEN 220
170  IF A$ = "E" THEN 200
180  IF A$ = "R" THEN 210
190  GOTO 140
200  &  RESTORE  GOTO 650: GOSUB
     290: GOTO 130
210  &  RESTORE  GOTO 1490: GOSUB
     290: GOTO 130
220  HGR2 : HCOLOR= 3: HPLOT 20,2
     4 TO 20,168 TO 279,168:YP =
     30:A$ = "E": GOSUB 580:YP =
     140:A$ = "I": GOSUB 580:XP =
     19:YP = 83
230  IF  PEEK (KBD) = 155 THEN  POKE
     CLK,0: GOTO 130
240  IF  PEEK (B0) < 128 AND  PEEK
     (B1) < 128 THEN 230
250  IF PEEK (B0) > 127 THEN YP =
     YP + 2 * (YP <  > 151): GOTO
     270
260 YP = YP - 2 * (YP <  > 15)
270  DRAW 4 AT XP,YP: IF XP < 271
      THEN XP = XP + 1: FOR X = 1
      TO 100: NEXT : GOTO 230
280  &  < :XP = 265: GOTO 230
290  HGR2 :YP = 7
300  READ A$: IF A$ = "*END*" THEN
      RETURN
310  IF  LEFT$ (A$,1) = "!" THEN
     J =  ASC ( MID$ (A$,2)) - 64
     :P$ =  MID$ (A$,3):P =  VAL
     (P$): ON J GOSUB 370,400,570
     ,570,410,420,440,450,570,510
     ,520,530,550,560: GOTO 300
320 A1$ =  RIGHT$ (A$,1): IF A1$ =
     "#" OR A1$ = "&" THEN B$ = A
     $:A$ = "": IF  LEN (B$) <  >
     1 THEN A$ =  LEFT$ (B$, LEN
     (B$) - 1)
330  IF A$ <  > "" THEN  GOSUB 58
     0
340  IF A1$ = "#" THEN  POKE CLK,
     0: GET A$: &  CLEAR SB,SR:YP
      = 0: IF  ASC (A$) = 27 THEN
     130
350  IF A1$ = "&" THEN  FOR X = 1
      TO 5000: NEXT
360  GOTO 300
370 PX = P:PY =  VAL ( MID$ (P$,5
     )): HCOLOR= 3:PR = 2: GOSUB
     610: GOSUB 620: GOSUB 630: GOSUB
     640
380 NX = 230:NY = 145: IF P = 140
      THEN NX = 65:NY = 100
390  RETURN
400 SR = P:SB =  VAL ( MID$ (P$,4
     )): &  CLEAR SB,SR:YP = 7: RETURN
410 PT = TN( VAL ( MID$ (P$,3))):
      FOR X = 1 TO P: & PT,20000 /
     PT: FOR Y = 1 TO 150: NEXT Y
     ,X: RETURN
420 YP = 50:A$ = "%%%%%%RELAX@@@"
430  GOSUB 580: HCOLOR= 0: FOR Y =
     186 TO 191: HPLOT 0,Y TO 279
     ,Y: FOR X = 1 TO 2200: NEXT
     X,Y: &  CLEAR SB,SR: POKE CL
     K,0:YP = 7: GOTO 560
440  &  CLEAR SB,SR:YP = 7:A$ = "
     REPEAT THE": GOSUB 580:A$ =
     "TENSION": GOSUB 580:A$ = "R
     ELAXATION": GOSUB 580:A$ = "
     CYCLE@ TENSE": GOSUB 580:A$ =
     "FIVE SECONDS@": GOTO 580
450 PR = 1: IF P = 0 THEN  HCOLOR=
     3: GOTO 470
460 BP = P: HCOLOR= 0
470  ON BP GOSUB 610,620,630,640
480  IF  LEN (P$) > 1 THEN  HCOLOR=
     0: FOR Y = NY TO NY + 14: HPLOT
     NX,Y TO NX + 28,Y: NEXT :A$ =
      MID$ (P$,2):XP = NX:YP = NY
     : GOSUB 590
490  IF P = 0 THEN YP = 7: &  CLEAR
     SB,SR: HCOLOR= 0: FOR Y = NY
      TO NY + 14: HPLOT NX,Y TO N
```

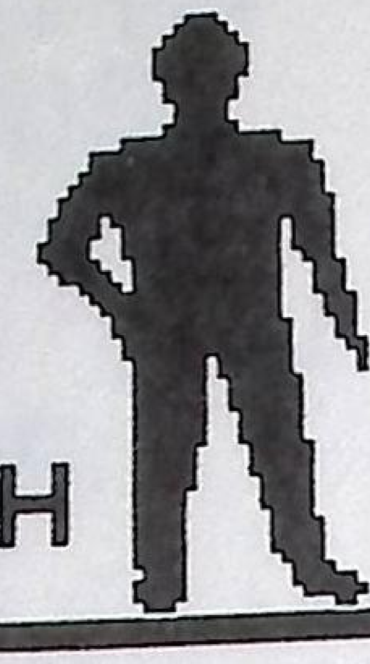



```
     X + 28,Y: NEXT
500 YP = 7: RETURN
510 YP = 50:HS = 18 -  LEN (P$):A
     $ = P$: FOR X = 1 TO HS:A$ =
     "^" + A$: NEXT : GOSUB 580: RETURN
520 A$ = "%%%%%%^TENSE": GOTO 540
530 A$ = "%%%%%%REPEAT"
540 YP = 50: GOSUB 580: GET A$: &
     CLEAR SB,SR: RETURN
550  &  CLEAR SB,SR:YP = 80:A$ =
     "%%%%RELAX": GOTO 430
560  HCOLOR= 3: FOR Y = 187 TO 19
     1: HPLOT 0,Y TO 279,Y: NEXT
     : RETURN
570  RETURN
580 XP = 0
590  HCOLOR= 0: HPLOT 0,YP: HCOLOR=
     3: FOR X = 1 TO  LEN (A$):AV
     =  ASC ( MID$ (A$,X,1)) - 6
     0: IF AV > 0 AND AV < 31 THEN
      DRAW AV AT XP,YP
600 XP = XP + 16 - 8 * (AV = 34):
      NEXT X:YP = YP + 21: RETURN
610  FOR Y = 1 TO PR: DRAW 31 AT
     PX,PY - Y: NEXT : RETURN
620  FOR Y = 1 TO PR: DRAW 32 AT
     PX + 3,PY + 39 - Y: NEXT : RETURN
630  FOR Y = 1 TO PR: DRAW 33 AT
     PX + 4,PY + 45 - Y: NEXT : RETURN
640  FOR Y = 1 TO PR: DRAW 34 AT
     PX + 17,PY + 69 - Y: DRAW 35
      AT PX + 19,PY + 90 - Y: NEXT
     : RETURN
650  DATA  !B29/186,!A246/30,!N,%
     ,FULL,RELAXATION,PROGRAM#
660  DATA  "THIS PROGRAM","IS DES
     IGNED","TO HELP YOU","LEARN
     TO",RELAX@#
670  DATA  PLEASE,"READ THE",DOCU
     MENTATION,BEFORE,ATTEMPING,"
     TO USE THIS",PROGRAM@#
680  DATA  %,%,"LET?S BEGIN@#"
690  DATA  "FIRST= LET US",BECOME
     ,FAMILIAR,"WITH THE",SCREEN@
     #
700  DATA  "ON THE RIGHT","SIDE O
     F THE","SCREEN THERE","IS A
     FIGURE@#"
710  DATA  "TEXT WILL","APPEAR TO
      THE","LEFT OF THE",FIGURE@#
720  DATA  "AT THE BOTTOM","OF TH
     E SCREEN","THERE IS A","TIME
     R BAR@#"
730  DATA  "THE FIRST FEW","TIMES
      THAT","YOU USE THIS","PROGR
     AM YOU","WILL NEED#"
740  DATA  "TO PAY CLOSE","ATTENT
     ION TO","THE TEXT@","AFTER A
      WHILE","YOU WILL",LEARN#
750  DATA THE,RELAXATION,INSTRUCT
     IONS,"AND YOU WILL","ONLY NE
     ED TO","LOOK AT THE#"
760  DATA  "FIGURE AND","LISTEN T
     O THE","AUDITORY CUES","TO D
     O THE",EXERCISES@#
770  DATA  "TO PREPARE=","IMAGINE
      A",TRANQUIL,SCENE@#
780  DATA  CONCENTRATE,"ON THIS",
     "SCENE AND"
790  DATA "ON YOUR","BREATHING SO
     ","IT IS DEEP","AND RHYTHMIC
     @#"
800  DATA  "CLOSE YOUR","EYES AND
      THEN","OPEN THEM","AGAIN AT
     ","THE TONE@#",!M
810  DATA  !E1/9,!H3%,"THE FIRST"
     ,"MUSCLE GROUP","YOU WILL","
     RELAX IS THE"
820  DATA  HANDS=,BICEPS=,"AND TR
     ICEPS",GROUP@#
830  DATA  "THIS TONE","WILL SIGN
     AL","YOU TO","PREPARE TO","R
     ELAX THIS","MUSCLE GROUP@#",
     !E1/8
840  DATA  !H3H,"BEGIN WITH","THE
      HANDS@","THIS TONE","SIGNAL
     S YOU","TO FOCUS","ON YOUR "
     ,HANDS@#,!E1/8,E1/7
850  DATA  "TENSE THE","HANDS AND
     ","WRISTS BY",SQUEEZING,YOUR
     ,"HANDS HARD@#"
860  DATA  "FEEL THE","TENSION IN
     ","YOUR FINGERS=","WRISTS= A
     ND","THE BACKS OF","YOUR HAN
     DS@#"
870  DATA "FEEL THE","TENSION FOR
     ","ABOUT FIVE",SECONDS@,"REL
     EASE THE",TENSION,SLOWLY@#
880  DATA  "ENJOY FEELING","RELAX
```




```
     ED@","AS THE BAR","GETS SMAL
     LER","LET YOURSELF#"
890  DATA  "BECOME","MORE AND MOR
     E","RELAXED@#",!M,!G,#,!E1/8
     ,!E1/7,!M
900  DATA  !H3B,"NOW YOU WILL","L
     EARN HOW TO","RELAX THE",BIC
     EPS@#
910  DATA  "THIS TONE","WILL SIGN
     AL","YOU TO RELAX","THE BICE
     PS@#",!E1/8,!E2/7
920  DATA  "THIS IS DONE","BY SQU
     EEZING","THE BICEPS","THE WA
     Y YOU","DO WHEN","SOMEONE SA
     YS#"
930  DATA  ">MAKE A",MUSCLE@>,"DO
      NOT MAKE","A FIST WITH","YO
     UR HANDS@#"
940  DATA  "BE SURE TO","REMEMBER
      TO","DO THIS WITH","BOTH AR
     MS@#"
950  DATA  "NOW LET THE","BICEPS
     RELAX@","PUSH >SPACE>","AND
     FOCUS ON","THE BAR@@@#",!M
960  DATA  !G,#,!E1/8,!E2/7,!M
970  DATA  !H3TF,"NOW YOU WILL","
     RS OF THE",FOREARMS@#
980  DATA  "THIS TONE","WILL SIGN
     AL","YOU TO","RELAX THE","TR
     ICEPS AND",UPPER,FOREARMS@#,
     !E1/8,!E3/7
990  DATA  "TO FEEL WHAT","IT IS
     LIKE TO","TENSE THESE","MUSC
     LES PRESS","YOUR FOREARMS#"
1000  DATA  "DOWNWARD INTO","THE
     ARMS OF A","CHAIR@ FEEL","TH
     E TENSION","FOR ABOUT","FIVE
      SECONDS@#"
1010  DATA  "NOW RELAX@@@","LET Y
     OURSELF","RELAX EVEN#"
1020  DATA  "MORE AS THE","BAR GE
     TS","SMALLER AND",SMALLER@,"
     PLEASE PRESS","THE SPACE",BA
     R@#
1030  DATA  !M,!G,#,!E1/8,!E3/7,!
     M
1040  DATA  !H,!H1%,"NOW YOU WILL
     ","LEARN HOW TO","RELAX THE"
     ,"FOREHEAD=","NOSE= AND",MOU
     TH@#
1050  DATA  "THIS TONE","WILL SIG
     NAL","YOU TO","PREPARE TO","
     RELAX THIS","MUSCLE GROUP@#"
     ,!E1/5
1060  DATA  !H1F,"THIS TONE","WIL
     L SIGNAL","YOU TO RELAX",THE
     ,FOREHEAD@#,!E1/5,!E1/6
1070  DATA  "TENSE THE","FOREHEAD
      BY","FROWNING","HARD@ HOLD"
     ,"THE FROWN","FIVE SECONDS@#
     "
1080  DATA  "RELAX THE","FROWN AN
     D","ENJOY THE",PLEASANT,SENS
     ATION@#
1090  DATA  !M,!G,#,!E1/5,!E1/6,!
     M
1100  DATA  !H1N,"THIS TONE","WIL
     L SIGNAL","YOU TO","PREPARE
     TO","RELAX THE",NOSE@#,!E1/5
     ,!E2/6
1110  DATA  "FIRST TENSE","THE MI
     DDLE OF","YOUR FACE BY","SCR
     UNCHING UP","THE NASAL",AREA
     @#
1120  DATA  "TRY TO KEEP","THE FO
     R EHEAD","SMOOTH@ TENSE","TH
     IS AREA","FOR ABOUT","FIVE S
     ECONDS@#"
1130  DATA  "NOW BEGIN","TO RELAX
     ","THIS AREA OF","YOUR FACE@
     @@#"
1140  DATA  !M,!G,#,!E1/5,!E2/6,!
     M
1150  DATA  !H1M,"THIS TONE","WIL
     L SIGNAL","YOU TO","PREPARE
     TO","RELAX THE","MOUTH AREA@
     #",!E1/5,!E3/6
1160  DATA  "TENSE THE","MOUTH BY
     ","PRESSING THE","LIPS TOGET
     HER","HARD@@@#"
1170  DATA  "HOLD THE","TENSION F
     OR","ABOUT FIVE",SECONDS=,"T
     HEN LET","YOURSELF@@@#"
1180  DATA  !M,!G,#,!E1/5,!E3/6,!
     M
1190  DATA  !H,!H2%,"NOW YOU WILL
     ","LEARN HOW TO","RELAX THE"
     ,NECK=,SHOULDERS=,"AND STOMA
     CH@#"
1200  DATA  "THIS TONE","WILL SIG
     NAL","YOU TO","PREPARE TO","
     RELAX THIS","MUSCLE GROUP@#"
     ,!E1/1
1210  DATA  !H2N,"THIS TONE","WIL
```

FOREHEAD, NOSE, AND MOUTH



```
L SIGNAL","YOU TO","PREPARE
TO","RELAX THE","NECK MUSCLE
S@#",!E1/1,!E1/2
1220  DATA  "TENSE THE","NECK BY"
,"PLACING A","HAND AGAINST",
YOUR,"FOREHEAD AND#"
1230  DATA  "PRESSING YOUR","HEAD
INTO","YOUR HAND@","DO THIS
FOR","FIVE SECONDS=",THEN@@
@#
1240  DATA  !M,!G,#,!E1/1,!E1/2,!
M
1250  DATA  !H2SH,"THIS TONE","WI
LL SIGNAL","YOU TO","PREPARE
 TO","RELAX THE","SHOULDERS@
#",!E1/1,!E2/2
1260  DATA  "TENSE THE","SHOULDER
S BY","PUSHING THEM","INTO Y
OUR","CHAIR@#"
1270  DATA  "HOLD THE","TENSION F
OR","ABOUT FIVE",SECONDS=,"T
HEN LET",YOURSELF@@@#
1280  DATA  !M,!G,#,!E1/1,!E1/2,!
M
1290  DATA  !H2ST,"THIS TONE","WI
LL SIGNAL","YOU TO","PREPARE
 TO","RELAX THE",STOMACH@#,!
E1/1,!E3/2
1300  DATA  "TENSE THE","STOMACH
BY","SUCKING IN","THE STOMAC
H",HARD@#
1310  DATA  "HOLD THE","TENSION F
OR","ABOUT FIVE",SECONDS=,"T
HEN LET",YOURSELF@@@#
1320  DATA  !M,!G,#,!E1/1,!E3/2,!
M
1330  DATA  !H,!H4%,"NOW YOU WILL
","LEARN HOW TO","RELAX THE"
,THIGHS=,CALVES=,"AND FEET@#
"
1340  DATA  "THIS TONE","WILL SIG
NAL","YOU TO","PREPARE TO","
RELAX THIS","MUSCLE GROUP@#"
,!E1/3
1350  DATA  !H4T,"THIS TONE","WIL
L SIGNAL","YOU TO","PREPARE
TO","RELAX THE",THIGH,MUSCLE
S@#,!E1/3,!E1/4
1360  DATA  "TENSE THE","THIGHS B
Y","STRETCHING","YOUR LEGS O
UT","IN FRONT OF#"
1370  DATA  "YOUR BODY AS","FAR A
S YOU","CAN@ DO THIS","FOR F
IVE","SECONDS THEN#"
1380  DATA  !M,!G,#,!E1/3,!E1/4,!
M
1390  DATA  !H4C,"THIS TONE","WIL
L SIGNAL","YOU TO","PREPARE
TO","RELAX THE","CALF MUSCLE
S@#",!E1/3,!E2/4
1400  DATA  "TENSE THE","CALVES B
Y","RESTING YOUR","HEELS ON
THE","FLOOR WITH#"
1410  DATA  "YOUR LEGS OUT","IN F
RONT OF","YOU@ NOW DRAW","YO
UR FEET","UP TOWARD#"
1420  DATA  "YOUR FACE@@@","FEEL
THE","TENSION AND","THEN LET
","THESE",MUSCLES@@@#
1430  DATA  !M,!G,#,!E1/3,!E2/4,!
M
1440  DATA  !H4F,"THIS TONE","WIL
L SIGNAL","YOU TO","PREPARE
TO","RELAX THE","FOOT MUSCLE
S@#",!E1/3,!E3/4
1450  DATA  "TENSE THE","FEET BY"
,"CURLING YOUR","TOES INWARD
","AND SQUEEZING#"
1460  DATA  "THEM HARD@","TENSE T
HE","FEET THIS WAY","FOR ABO
UT","FIVE SECONDS@",THEN@@@#
1470  DATA  !M,!G,#,!E1/3,!E3/4,!
M
1480  DATA  &,*END*
1490  DATA  !B39/77,!A140/80,!N,"
REVIEW SEQUENCE#"
1500  DATA  "HANDS= BICEPS=","%AN
D TRICEPS@",!E1/8,#
1510  DATA  !JHANDS,!H3H,!E1/8,!E
1/7,#,!K,!F
1520  DATA  !L,!E1/8,!E1/7,!K,!F
1530  DATA  !JRICEPS,!H3B,/,!E1/8
,!E2/7,#,!K,!F
1540  DATA  !L,!E1/8,!E2/7,!K,!F
1550  DATA  !JTRICEPS,!H3T,/,!E1/
8,!E3/7
1560  DATA  #,!K,!F
1570  DATA  !L,!E1/8,!E3/7,!K,!F
1580  DATA  !H,"%%%FOREHEAD=","NO
SE= AND MOUTH",!E1/5,#
1590  DATA  !JFOREHEAD,!H1F,!E1/5
,!E1/6
1600  DATA  #,!K,!F,!L,!E1/5,!E1/
```

THIS PROGRAM
IS DESIGNED
TO HELP YOU
LEARN TO
RELAX.



```
     6,!K,!F
1610  DATA   !JNOSE,!HIN,!E1/5,!E2
     /6
1620  DATA  #,!K,!F,!L,!E1/5,!E2/
     6,!K,!F
1630  DATA   !JMOUTH,!H1M,!E1/5,!E
     3/6
1640  DATA  #,!K,!F,!L,!E1/5,!E3/
     6,!K,!F
1650  DATA  !H,"NECK= SHOULDERS="
     ,"AND STOMACH@",!E1/1,#
1660  DATA  !JNECK,!H2N,!E1/1,!E1
     /2
1670  DATA  #,!K,!F,!L,!E1/1,!E1/
     2,!K,!F
1680  DATA  !JSHOULDERS,!H2SH,!E1
     /1,!E2/2
1690  DATA  #,!K,!F,!E1/1,!E2/2,!
     K,!F
1700  DATA  !JSTOMACH,!H2ST,!E1/1
     ,!E3/2
```

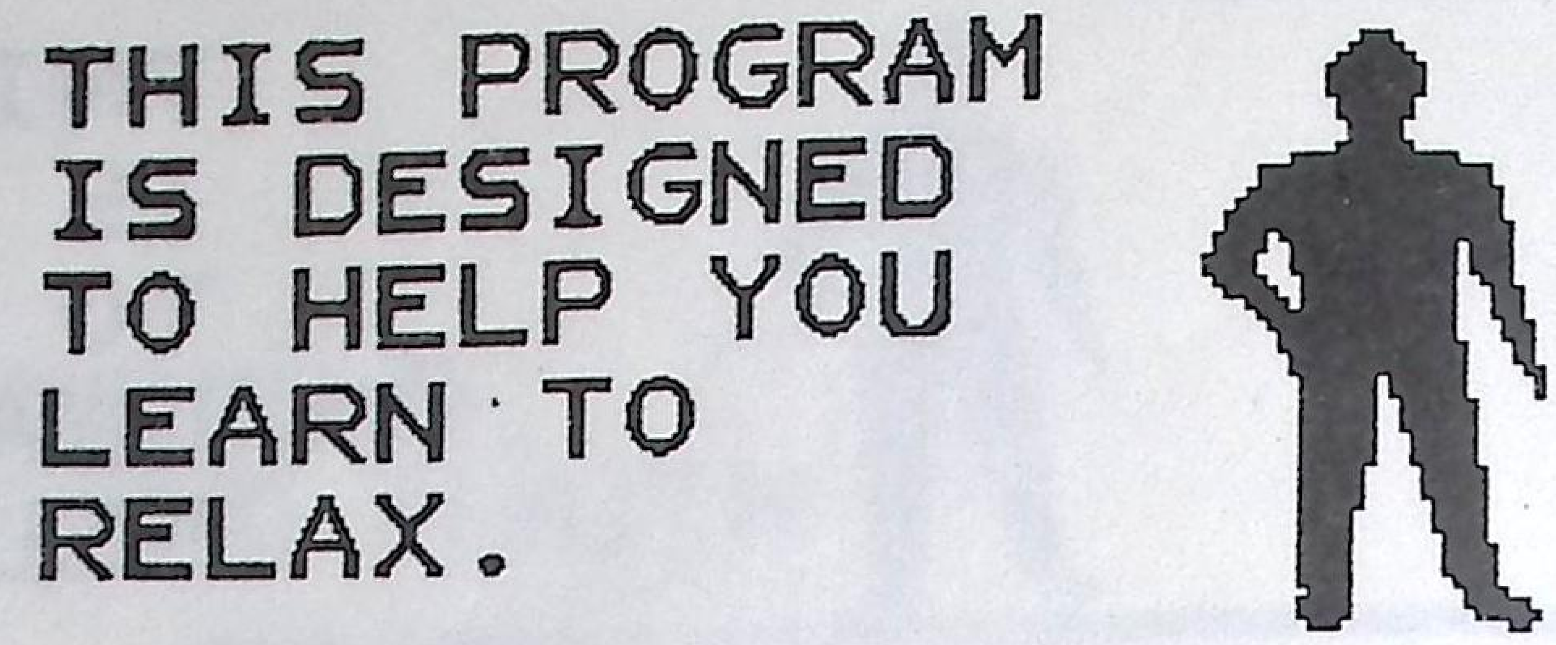


```
1710  DATA  #,!K,!F,!L,!E1/1,!E3/
     2,!K,!F
1720  DATA  !H,"THIGHS= CALVES=",
     "%%%AND FEET@",!E1/3,#
1730  DATA  !JTHIGHS,!H4T,!E1/3,!
     E1/4
1740  DATA  #,!K,!F,!L,!E1/3,!E1/
     4,!K,!F
1750  DATA  !JCALVES,!H4C,!E1/3,!
     E2/4
1760  DATA  #,!K,!F,!L,!E1/3,!E2/
     4,!K,!F
1770  DATA  !JFEET,"!H4F",!E1/3,!
     E3/4
1780  DATA  #,!K,!F,!L,!E1/3,!E3/
     4,!K,!F
1790  DATA  !H,&,*END*
1800  DATA  %%%^RELAXATION,%%%%%T
     RAINER,%,"R%%REVIEW^SEQUENCE
     ","E%%ENTIRE PROGRAM","D%%DE
     EP BREATHING",%%%TRAINER,%,"
     ENTER CHOICE",*END*
```

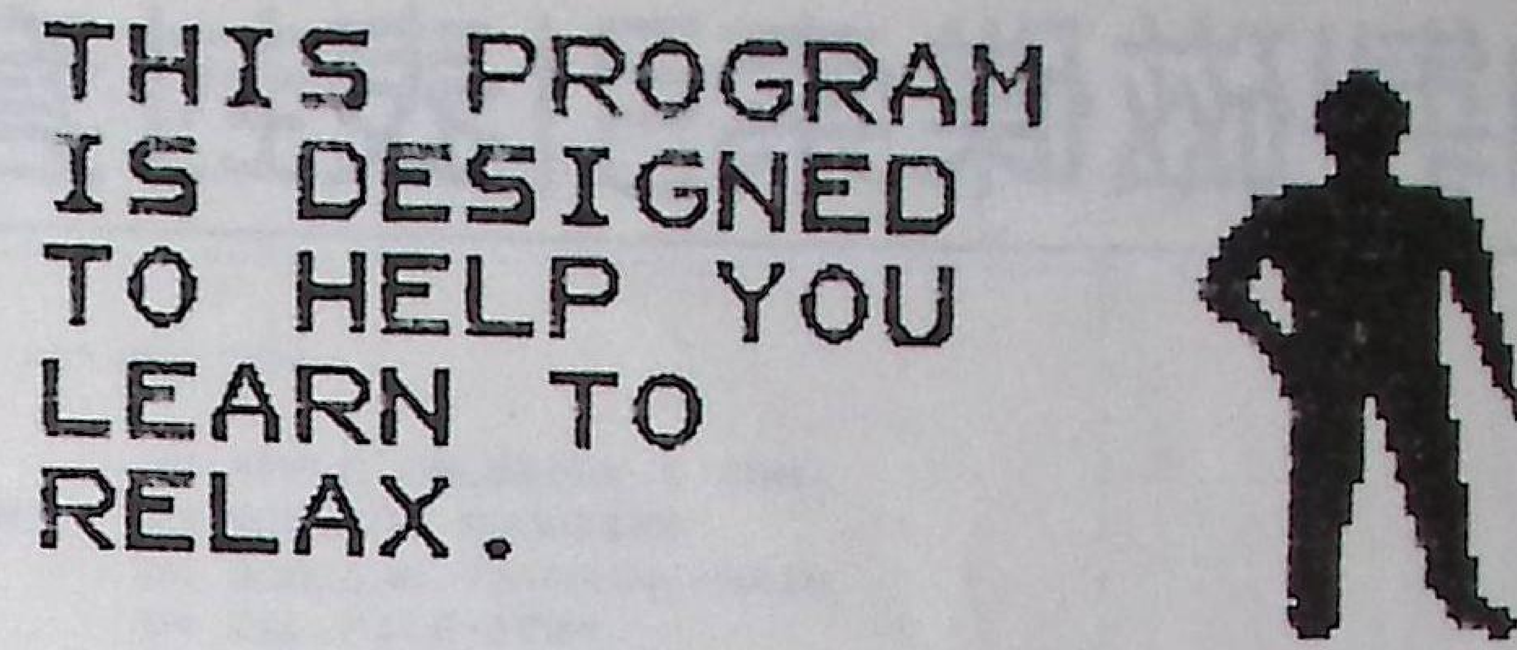
CREAT RELAX·FILES 副程式

```
]LIST

10  HIMEM: 24576:D$ =  CHR$ (4):L
     = 0
20  READ N: IF N > 255 THEN  PRINT
     D$"BSAVE RELAX ROUTS,A$300,L
     "L:L = 0: GOTO 40
30  POKE 768 + L,N:L = L + 1: GOTO
     20
40  READ N: IF N > 255 THEN  PRINT
     D$"BSAVE RELAX TEXT,A$6000,L
     "L:L = 0: END
50  IF N >  = 0 THEN  POKE 24576 +
     L,N:L = L + 1: GOTO 40
60  READ N2: FOR X = 1 TO  - N: POKE
     24576 + L,N2:L = L + 1: NEXT
     : GOTO 40
100  DATA 201,189,208,33,32,245,2
     30,134,206,32,76,231,134,207
     ,165,206,32,17,244,169,0,164
     ,207,145,38,136,16,251,198,2
     06,165,206,201,255,208,236,9
```




```
    6,201,209,208,36,32,177,0,16
    9,167,133,206,32,17,244,160,
    3,200,177,38,136,145,38
110 DATA  200,192,39,208,245,169
    ,0,145,38,198,206,165,206,20
    1,23,208,228,96,201,174,208,
    51,32,177,0,169,171,32,192,2
    22,32,12,218,32,26,214,176,3
    ,76,124,217,160,4,177,155,20
    1,131,240,3,76,188,225,152,2
    4,101,155,133,125,165,156
120 DATA 105,0,133,126,165,80,13
    3,123,165,81,133,124,96,32,2
    48,230,232,134,206,32,190,22
    2,32,103,221,32,82,231,14,88
    ,255,44,48,192,32,200,3,44,4
    8,192,32,200,3,44,48,192,32,
    200,3,32,200,3,44,48,192,32,
    200,3,32,200,3,56,165
130 DATA  80,233,1,133,80,165,81
    ,233,0,133,81,208,208,165,80
    ,208,207,96,166,206,37,0,202
    ,208,251,96,256
200 DATA  35,0,120,0,137,0,148,0
    ,160,0,177,0,227,0,41,1,94,1
    ,151,1,217,1,9,2,67,2,130,2,
    170,2,211,2,10,3,49,3,114,3,
    179,3,235,3,28,4,86,4,150,4,
    223,4,7,5,64,5,108,5,164,5,2
    13,5,2,6,65,6,137,6,148,7,85
    ,8,148,9,38,10,-46,0,-4,146
210 DATA  74,73,45,53,63,55,45,5
    3,63,46,30,7,0,73,54,46,36,1
    08,9,54,46,36,44,0,73,41,45,
    62,63,46,45,62,55,245,7,0,-4
    ,146,82,73,41,45,56,63,39,-2
    ,45,28,63,7,0,-3,146,50,54,4
    6,37,39,37,39,37,39,12,-3,10
    0,44,53,21,-4,118,-2,54
220 DATA  63,44,60,44,-2,60,-3,6
    3,60,12,44,36,37,44,116,54,5
    3,46,54,253,-2,63,7,0,-7,54,
    -5,45,5,-2,40,32,36,-2,28,12
    ,36,-3,60,-4,63,-6,54,46,-4,
    45,-2,44,36,-2,60,-3,63,39,-
    3,45,-2,37,60,-2,56,-3,63,-5
    ,54,46,-3,45,5,0,73,9,-2,45
230 DATA  -3,21,-2,30,100,-2,39,
    -2,63,23,-2,55,-3,54,-3,46,-
    2,45,44,36,172,21,55,-2,30,-
    3,63,-3,60,-4,36,-3,37,53,15
    1,155,-3,54,6,0,-7,54,-5,45,
    -2,40,32,-3,36,228,-2,28,-4,
    63,-6,54,46,-3,45,-3,37,-3,3
    6,-3,60,-3,63,-5,54,46
240 DATA  -3,45,0,-6,45,54,-5,63
    ,54,46,-2,45,53,62,60,62,60,
    62,60,-3,54,-5,45,54,-6,63,-
    6,36,44,-5,45,222,-2,219,27,
    -5,54,46,-5,45,0,-6,45,54,39
    ,55,39,55,39,55,39,55,39,55,
    54,46,-2,45,53,62,60,62,60,6
    2,60,-4,54,63,-6,36,44
250 DATA  -6,54,6,0,73,9,-3,45,4
    6,62,39,-3,63,191,23,246,-2,
    54,14,118,14,173,-2,45,-4,37
    ,36,60,-2,63,55,-2,45,53,62,
    190,23,-2,63,28,-2,39,60,-2,
    36,37,-2,44,-2,45,173,-2,219
    ,155,18,54,6,0,45,-3,54,-4,4
    5,-3,36,45,-7,54,63,-3,36
260 DATA  -4,63,-3,54,63,-6,36,4
    4,-3,54,-5,45,-3,54,223,-2,2
    19,35,-2,36,68,-3,73,-3,36,0
    ,9,-4,45,62,-3,63,119,41,45,
    245,63,-4,54,-2,62,55,45,37,
    53,-2,45,60,39,55,-4,36,60,-
    4,54,6,0,-2,73,9,-2,45,62,63
    ,119,45,-4,54,-4,62,-2,63
270 DATA  -3,56,32,44,54,-3,53,4
    5,101,12,44,-3,36,60,-3,54,6
    ,0,-7,54,37,-6,36,44,-4,54,4
    6,-9,37,22,-5,30,-5,46,54,7,
    -5,56,184,30,63,14,55,54,5,0
    ,45,-6,54,-5,45,54,-6,63,-6,
    36,44,-6,54,-5,45,5,0,-7,54,
    37,-6,36,44,-2,54,-4,53
280 DATA 5,32,-3,37,36,44,53,55,
    53,55,53,63,46,53,-3,54,62,3
    9,-3,36,53,-2,54,254,218,219
    ,27,-5,36,8,-3,21,-3,12,4,0,
    -7,54,37,-6,36,44,54,-8,46,-
    2,54,45,-7,36,63,-4,54,46,-4
    ,36,150,-2,146,54,254,8,24,7
    ,-6,56,48,-4,54,46,0,73
290 DATA 9,45,85,-3,63,-2,23,246
    ,-2,54,14,118,14,-3,45,213,2
    7,63,196,35,39,60,-2,36,37,-
    2,44,-3,45,21,63,14,173,63,4
    6,53,63,46,53,63,46,-2,62,18
    8,-2,63,68,24,216,36,4,0,-7,
    54,37,-6,36,44,-4,45,-2,21,-
    2,54,-2,55,-3,63,39,-3,45
300 DATA  -2,12,36,-2,60,-3,63,5
```




```
     5,-6,54,5,0,73,9,45,85,-3,63
     ,62,23,246,-2,54,14,118,21,1
     73,45,5,-2,40,-2,46,4,192,36
     ,37,36,60,36,39,-3,63,30,55,
     62,-2,54,53,-2,46,53,45,60,1
     3,12,37,100,39,37,39,228,223
     ,219,146,51,54,0,-7,54
310  DATA  37,-6,36,44,-4,45,-3,2
     1,54,246,30,-4,63,44,-3,45,-
     2,37,36,-2,39,-4,63,-3,54,11
     8,73,-5,53,45,-5,28,223,219,
     -2,54,46,0,-2,73,-3,45,54,-4
     ,63,-2,55,46,21,-3,45,-2,46,
     -2,54,-2,62,-5,63,36,-4,45,-
     2,37,60,56,-3,63,-2,60
320  DATA  -2,36,-2,37,45,117,73,
     9,-4,63,-2,30,54,21,53,-3,45
     ,-2,14,54,-2,62,-4,63,7,0,-6
     ,45,54,-2,63,55,-5,54,62,39,
     -5,36,60,-2,63,44,-5,45,223,
     155,-6,54,0,45,-5,54,-3,46,4
     5,-3,37,-5,36,45,-6,54,-2,55
     ,-4,63,-2,60,-5,36,44,-5,54
330  DATA  -2,14,77,73,-2,12,-5,3
     6,4,0,-2,54,53,46,54,53,46,5
     4,53,46,37,44,36,37,44,36,37
     ,44,-2,36,63,46,62,46,30,62,
     54,55,62,54,55,38,24,36,39,6
     0,36,39,60,36,44,54,6,0,-4,5
     4,53,46,54,53,46,5,32,36,44,
     172,62,46,62,-2,14,37,44
340  DATA  36,37,44,-4,36,63,-4,5
     4,190,154,62,223,36,192,35,-
     4,36,55,-3,54,77,-2,73,33,-3
     ,36,0,-6,46,54,-6,55,45,44,-
     4,40,-3,46,54,45,60,228,-3,2
     8,220,13,44,32,-4,37,63,190,
     -2,23,62,-4,28,60,7,0,54,-5,
     53,-3,54,46,37,-3,36,-5,44
350  DATA  36,63,46,30,-3,23,55,-
     3,54,38,8,24,8,24,8,-2,24,-3
     ,28,36,55,6,0,-6,45,53,62,-6
     ,63,44,-5,45,53,94,59,-9,62,
     55,46,-6,45,36,55,39,55,39,5
     5,39,55,39,55,39,23,39,96,-7
     ,12,4,0,-2,45,109,18,-4,63,2
     55,18,-6,45,109,18,-8,63
360  DATA  183,-8,45,181,-8,63,18
     3,9,-6,45,181,-6,63,183,9,-4
     ,45,181,-5,63,0,8,24,8,24,8,
     24,8,24,8,24,8,24,8,24,8,24,
     8,24,216,219,27,-6,45,109,-2
     ,73,18,-14,63,255,219,147,-2
     0,45,181,27,-18,63,183,-18,4
     5,181,27,-16,63,183,-16,45
370  DATA  181,-16,63,183,-16,45,
     181,27,-14,63,183,-14,45,181
     ,-14,63,183,9,-13,45,181,-13
     ,63,183,-14,45,0,8,24,8,24,8
     ,24,8,24,8,24,8,24,8,24,8,24
     ,8,24,216,-6,219,59,159,42,4
     5,141,58,-3,63,159,42,-4,45,
     22,-4,63,255,18,-4,45,181
380  DATA  -4,63,255,18,-4,45,181
     ,-4,63,183,9,-4,45,109,18,-4
     ,63,183,73,41,-3,45,141,58,-
     3,63,86,41,-2,45,22,63,183,4
     5,181,183,173,146,-14,73,37,
     56,31,8,24,45,37,216,63,39,4
     0,45,4,56,-2,63,4,40,-2,45,1
     96,59,-2,63,4,40,-2,45
390  DATA  4,56,-2,63,31,8,24,-2,
     45,37,56,-3,63,4,40,-3,45,4,
     56,-3,63,4,40,-3,45,4,56,-3,
     63,4,40,-2,45,4,56,-2,63,68,
     41,45,196,59,7,0,8,24,8,24,8
     ,24,8,24,8,24,8,24,8,24,8,24
     ,8,24,56,-13,63,22,-13,45,18
     1,-13,63,183,-14,45,181
400  DATA  -14,63,183,-14,45,181,
     -6,63,255,59,-6,63,22,-6,45,
     109,41,-6,45,141,58,-6,63,22
     3,27,-6,63,183,-6,45,109,73,
     -6,45,181,-6,63,255,219,-6,6
     3,183,-6,45,109,73,9,-5,45,1
     81,-5,63,255,219,27,-6,63,18
     3,-6,45,109,73,9,-5,45
410  DATA  109,18,-5,63,255,-3,21
     9,-5,63,183,-5,45,109,-3,73,
     -5,45,181,-5,63,255,-3,219,-
     5,63,183,-5,45,109,-3,73,9,-
     4,45,181,-4.63,255,-4,219,59
     ,-4,63,22,-4,45,109,-4,73,41
     ,-4,45,5,0,73,57,-3,63,223,-
     4,219,27,-4,63,183,-4,45
420  DATA  109,-5,73,-4,45,181,-4
     ,63,255,-5,219,-4,63,183,-4,
     45,109,-5,73,-4,45,181,-3,63
     ,255,-5,219,27,-4,63,183,9,-
     3,45,109,-6,73,41,-3,45,77,1
     45,-5,63,255,-6,219,59,-4,63
     ,22,-4,45,109,-7,73,-5,45,18
     1,27,63,255,-9,219,27,-3,63,
     0,256
```

自製磁碟索引表封套

本程式能將磁碟內的索引表貯入記憶內，並將之印在一個自製封套上，有了本程式後，便可以對磁碟的內容一目了然，無須再開機找尋了。

使用本程式十分容易，只須開 PRINTER（PRINTER INTERFACE 插在 SLOT 1上）和按下 RETURN 便行了。

今期電腦時代程式磁碟將隨碟附送一個 DISK NO. #3 的索引磁碟封套，各位讀者在購買時請留意。封套將會跟碟一齊膠封。□

自製磁碟索引表封套程式

```
1   REM  ************************
2   REM  *                      *
3   REM  *   DISK DIRECTORY     *
4   REM  *    BY COMPUHOUSE     *
5   REM  *                      *
6   REM  ************************
50  TEXT : HOME :IND = 0
60  VTAB 3: HTAB 5: FOR I = 1 TO
     32: PRINT  CHR$ (95);: NEXT
    : PRINT
70  PRINT  TAB( 5)"!"; TAB( 36);"
    !": PRINT  TAB( 5)"!"; TAB(
    36);"!"
80  VTAB 6: HTAB 5: PRINT "!
     DISK DIRECTORY          !":
    PRINT  TAB( 5)"!"; TAB( 36)
    ;"!"
90  FOR I = 1 TO 40: PRINT "-";: NEXT
100  FOR I = 1 TO 3: PRINT "!"; TAB(
    40);"!";: NEXT
110  VTAB 12: PRINT "! IT'S EASY'
     JUST SET UP YOUR PRINTER, !
    ! PUT A DISKETTE IN YOUR DIS
    K DRIVE    !!         AND P
    RESS RETURN.            !";
120  FOR I = 1 TO 4: PRINT "!"; TAB(
    40);"!";: NEXT I
130  FOR I = 1 TO 20: PRINT "--";
    : NEXT I
140  POKE 1920,171: POKE 1959,171
    : POKE 1924,186: POKE 1955,1
    86: POKE 1360,171: POKE 1399
    ,171
150  VTAB 22: HTAB 20: GET GO$: PRINT
    : HTAB 20: VTAB 23
160  LOMEM: ( PEEK (106) + 17) *
    256
170 Z =  PEEK (106) - 16
180 D$ =  CHR$ (4):DD$ = "-------
    ---------------------------
    ---------------------------
    --------":LL$ = "!      !":L
    R$ = LL$
190  PRINT D$;"NOMON I,0,C"
195  IF IND THEN  POKE 54,C - 296
    6 -  INT (C / 256): POKE 55,
     INT ((C - 2966) / 256): GOTO
    240
200  FOR I = 768 TO 809: READ K: POKE
    I,K: NEXT I
210  POKE 6,0: POKE 7,Z
220 C =  PEEK (54) +  PEEK (55) *
    256
230 C = C + 2966
240  POKE C,0: POKE C + 1,3
250  POKE C + 2,39: POKE C + 3,3
260  PRINT D$;"CATALOG"
270  PRINT D$;"PR#0": PRINT D$"IN
    #0"
280  POKE 54,0: POKE 55,193
290 M =  PEEK (111) +  PEEK (112)
     * 256:M = M + 7600
300 C = 1
310  PRINT : PRINT DD$;"  CUT": PRINT
    LL$; SPC( 55);LR$
320  PRINT LL$; SPC( 20);
330  FOR N = M + 11 TO M STEP  -
    1: PRINT  CHR$ ( PEEK (N) -
    128);: NEXT N
340  IF  PEEK (47094) < 100 THEN
     PRINT " ";
350  IF  PEEK (47094) < 10 THEN  PRINT
    " ";
360  PRINT  STR$ ( PEEK (47094));
     SPC( 20);LR$: PRINT LL$; SPC(
    55);LR$
370  FOR I = Z * 256 + 19 TO ( PEEK
    (6) +  PEEK (7) * 256) - 1 STEP
    76
381 K = I + 38
390  PRINT LL$"  ";
400  FOR J = I + 4 TO I + 28
410  PRINT  CHR$ ( PEEK (J));
420  NEXT J:C = C + 1
430  IF K > (( PEEK (6) + 256 *  PEEK
```

```
----------------------------------------------------------------------- CUT
!          !                                                     !         !
!          !                 DISK VOLUME 254                     !         !
!          !                                                     !         !
!          ! 51 APPLE CALENDAR          03 APPLE CALENDAR & SUM   !         !
!          ! 03 APPLESOFT TO FORTRAN    20 BEND OF ROUNDING       !         !
!          ! 03 BOOLEAN TUTORIAL        04 BOOLEAN TUTORIAL-MAIN  !         !
!          ! 04 CAL.FILE-1983           04 CAL.FILE-1984          !         !
!          ! 04 CAL.FILE-1985           04 CAL.FILE-1986          !         !
!          ! 36 COMPUTEXT               02 DVORAK KEYBOARD        !         !
!          ! 08 DYNAMIC CALENDER        10 GAME OF CONNECTING FOU !         !
!          ! 09 GAME OF MATHEMATIC      27 GAME OF MAZING         !         !
!          ! 18 GAME OF SPELLING TEST   06 GREETING               !         !
!          ! 09 GREETING TRANSFER       06 MAZE SHAPES            !         !
!          ! 02 PEELING LISTING 5       06 PEELING THE TEXT SCREE !         !
!          ! 05 RETRIEVE THE DELETED P  34 SUM1                   !         !
!          ! 02 TEXT TEST               23 GAME OF SABOTAGE       !         !
!          ! 02 HISAB                                             !         !
!          !                                                     !         !
!          !                                                     !         !
----------------------------------------------------------------------- FOLD
           !                                                     !
           !                                                     !
           !                                                     !
           \-----------------------------------------------------/  CUT
```

各位如欲索取「電腦時代程式磁碟第一及第二期」索引表封套，可連同回郵信封（貼四角郵票）寄九龍中央郵政信箱71193號，電腦時代出版社收。信封面請註明「索取」。

```
    (7)) - 1) THEN  PRINT  SPC(
    28);LR$: GOTO 510
440  PRINT " ";
450  FOR J = K + 4 TO K + 28
460  PRINT  CHR$ ( PEEK (J));
470  NEXT J
480  PRINT " "LR$;: PRINT
490  IF C = 20 THEN LR$ = "!
       ":LL$ = "       !": PRINT
    DD$;"  FOLD": PRINT LL$: SPC(
    55);LR$ PRINT LL$; SPC( 55);
    LR$:C = C + 2
500  NEXT I
510  FOR I = 1 TO 48 - C
520 C = C + 1
530  IF C = 20 THEN LR$ = "!
       ":LL$ = "       !": PRINT
    DD$;"  FOLD": PRINT LL$; SPC(.
    55);LR$: PRINT LL$; SPC( 55)
    ;LR$:I = I + 2
540  PRINT LL$; SPC( 55);LR$: NEXT
     I
550  PRINT "          "; CHR$ (92);"
     -----------------------------
     ----------------------------/
       CUT": PRINT  CHR$ (12)
560  PR# 0
570  HOME : VTAB 12: HTAB 6: PRINT
     "ENVELOPE ANOTHER DISKETTE?"
     ;: GET Q$: PRINT :IND = 1
580  IF  LEFT$ (Q$,1) < > "N" THEN
     50
590  DATA  56,72,140,38,3,160,0,2
     01,32,176,2,105,192,201,96,1
     76,2,105,128,201,128,176,2,1
     05,64,145,6,230,6,208,2,230,
     7,104,172,38,3,96,0,169,160,
     96
600  END
```

雜誌目錄分類歸檔紀錄系統

讀者們有否曾經試過在找尋某篇所需文章時，一時之間，想不起文章在那本雜誌之內。或者知道在那本雜誌，但却不知文章所在的頁數。如果確曾有此經驗，本文就絕對不能錯過。筆者在此，介紹一個雜誌目錄索引系統，它用來將文章分類和編入目錄，並能隨時查閱，刪改或取消。而且，用家更能爲每本雜誌開創個別檔案。

程式運行

在輸入好程式，存貯及按入 RUN 後，螢幕會出現系統介紹，只需按下任何鍵掣便可獲得主要八項索引系統的目錄選擇：

（1） INITIALIZE A FILE（開檔案）
（2） LOAD ANOTHER MAGAZINE（載入雜誌名稱）
（3） ADD ARTICLES（加插文章名稱）
（4） CHANGE ARTICLES（更改文章名稱）
（5） DELETE ARTICLES（刪除文章名稱）
（6） LIST AN ARTICLE（列出文章名稱）
（7） FIND／SORT ARTICLES（尋找需要之文章）
（8） EXIT THE PROGRAM（離開程式）

由於這是第一次接觸這個程式，故應先選擇 OPTION 1。在這個項目下，用家可輸入多至 6 個的標題（次序由用家自定）。如果不輸入任何資料而按 RETURN 鍵，螢幕會再次顯示可供選擇之項目。不過，假如有輸入標題才按 RETURN 掣，標題會全部列出。萬一標題輸入錯誤，可按“N”鍵來再次輸入。

1.開檔案

當選擇開檔案後，螢幕會作如下的顯示：

（1） ARTICLE TITLE（文章題目）
（2） VOLUME NUMBER （卷號）
（3） ISSUE NUMBER （期號）
（4） PAGE NUMBER （頁數）
（5） CATEGORY （目錄）
（6） COMMENTS （備註）

現在，用家可輸入每個標題下之數目位數。 VOLUME 及 ISSUE 號數用兩位數已足夠，頁數則最好預定有三位數字。此外，要留意預備有足夠空位作加入附註及題目之用。不過，所加之句子切忌太長，因會佔用磁碟空間。

做好以上輸入後，系統會問用家輸入之預定位數是否正確。如果正確，輸入雜誌名稱及按 RETURN 鍵，系統程式便完成了在磁碟內之資料分類檔案。

2.載入雜誌名稱

這功能是用於編輯另外一個雜誌檔案，只需輸入需要的雜誌名稱或輸入符號（*）來將磁碟編目錄。

3.加插文章名稱

這用來輸入所選擇標題下的資料。用家不能輸入多過500篇文章。限制文章數目原因是磁碟空間有限。在讀後操作指示，可按任何鍵。螢幕上出現第一個標題及先前用家所預定數字位數刻痕。

當輸入 VOLUME、ISSUE，PAGE等號碼時，要在適當位置補 0。例如 VOLUME 3，ISSUE 8，PAGE NUMBER 45，應寫爲 03，08，045。 如果用家嘗試輸入比指定的數目爲多時，鈴聲會响，而號碼亦不會印出。在第一個標題後按 RETURN鍵，或在按R鍵時同時按 CTRL 鍵，效果會使螢幕重現 MENU OPTIONS。

註：下面假子中 RECORD指一雜誌文章。

```
RECORD #1          :
ARTICLE TITLE      : POKER SHOWHAND
VOLUME NUMBER      : 01
ISSUE NUMBER       : 02
PAGE NUMBER        : 018
CATEGORY           : GAME
COMMENTS           : ONE TO FOUR PLAYERS
```

輸入上面資料，系統會詢問是否正確，如答是的話（按Y）系統程式，會存貯起資料，否則會洗去所有資料，而用家可重新開始。

4.更加文章名稱

這功能可將磁碟內記錄改變，或只更改某一紀錄，或將資料補充。按任何掣可得到第一個紀錄。如想改變紀錄，按入“Y”後，游標會在第一個紀錄開端出現。按 RETURN 會保存資料及將游標移至下一標題。如想改正資料，只需按入正確資料才按 RETURN。在審閱全部標題之後，電腦會問資料正確否。如爲正確，便會被存入磁碟。

5.刪除文章名稱

這選擇會問用家希望取消的紀錄。輸入紀錄名稱後，電腦會再次要求證實。只需打入“YES”，電腦便將紀錄內資料取消，用檔案內最後記錄代入。

6.列出文章名稱

利用這功能，可列出某一紀錄或所有紀錄和用螢幕或印字機印出。在印出個別紀錄後，電腦會問用家繼續或停止。但按 RETURN鍵，會重返 MENU OPTIONS。

7.尋找需要之文章

這選擇可找出需要文章內的某一標題下內容。螢幕上會顯示：

（1） ARTICLE TITLE （文章題目）
（2） VOLUME NUMBER （卷號）
（3） ISSUE NUMBER （期號）
（4） PAGE NUMBER （頁數）
（5） CATEGORY （目錄）
（6） COMMENTS （備註）

至於 SORT 的功能，能將磁碟內所有紀錄據某一標題分類。按入分類標題後待〝SORT COMPLETED〞出現螢幕，然後用家可翻看分類列出。在查看全部紀錄或按RETURN掣，電腦會問用家需否存貯起資料。如打入〝Y〞，系統程式，會存起分類後之檔案。

本系統程式所用之存貯系統是一個隨機存取檔案，它的內容只會進入一個紀錄。這使用家容易存取、改變、分類或找尋任何檔案。

行號 3200 起，是予加及改變選擇用的子程式。這子程式抑制所有控制鍵及游標鍵的功能。唯一例外的是用來編輯的右邊箭咀。

系統程式的另外一個特性是分類功能。在用家指定要分類的標題後，系統便在每一個記錄中閱讀這個標題下的內容。筆者在程式中用一個 FOR…… NEXT 迴路於行號7580中來讀N$（1），H次數（H是標題號數）。這表示會檢讀標題直至需要的找到爲止。

結論

筆者相信這個雜誌索引分類歸檔系統，對用家非常有用。如果用家想修改程式，可將某一檔案標題更改及改變行號的資料句子。此外，這系統可令用家能創造自己的圖書館檔案。□

```
 10  REM APPLE M.A.T.
 15  REM MAGAZINE ARTICLE TRACKING
 20  ONERR  GOTO 8000
 25  FOR I = 1 TO 6: READ HE$(I): NEXT
 30  DATA   ARTICLE TITLE,VOLUME N
     UMBER,ISSUE NUMBER,PAGE NUMB
     ER,CATEGORY,COMMENTS
 40  TEXT : NORMAL : SPEED= 255:D$
      =  CHR$ (4):G$ =  CHR$ (7)
 43  IF  PEEK (768) = 29 THEN 100
 45  HOME
 57  VTAB 5: HTAB 11: INVERSE : PRINT
     "A P P L E  M. A. T.": NORMAL
 60  VTAB 7: HTAB 3: PRINT "(MAGAZ
     INE ARTICLE TRACKING SYSTEM)
     "
 90  VTAB 18: HTAB 8: PRINT "PRESS
     ANY KEY TO CONTINUE";: GET
     A$
 100  HOME : HTAB 3: INVERSE : PRINT
      "MAGAZINE ARTICLE TRACKING S
      YSTEM": VTAB 4: HTAB 13: FLASH
      : PRINT "MENU OPTIONS": NORMAL
 110  VTAB 6: PRINT  TAB( 7)"1. IN
      ITIALIZE": PRINT : PRINT  TAB(
      7)"2. LOAD": PRINT : PRINT  TAB(
      7)"3. ADD"
 120  PRINT : PRINT  TAB( 7)"4. CH
      ANGE": PRINT : PRINT  TAB( 7
      )"5. DELETE": PRINT : PRINT
      TAB( 7)"6. LIST"
 130  PRINT : PRINT  TAB( 7)"7. FI
      ND/SORT": PRINT : PRINT  TAB(
      7)"8. EXIT"
 140  VTAB 23: PRINT "WHICH OPTION
       WOULD YOU LIKE ?";
 150  GET A$:A =  VAL (A$): IF A <
      1 OR A > 8 THEN 150
 155  ON A GOSUB 1000,2000,3000,40
      00,5000,6000,7000,59999
 160  GOTO 100
 200 FL = 0: IF MN$ = "" THEN FL =
      1
 210  IF DA$(1) = "" THEN FL = 1
 220  IF NU = 0 THEN FL = 1
 230  IF FL = 1 THEN  HOME : GOSUB
      2010
 240  IF  LEN (MN$) = 0 THEN  POP
      : RETURN
 250  RETURN
 1000  HOME :A$ = "INITIALIZE FILE
      S": GOSUB 60000: PRINT : PRINT
      : PRINT "ENTER THE HEADINGS
      FOR THE DATA BASE:"
 1020 NU = 1: VTAB 7: FOR I = 1 TO
      6: PRINT  TAB( 7)I". "HE$(I)
      : PRINT : NEXT
 1040  VTAB 21: PRINT "WHAT IS HEA
      DING #"NU" ?";: GET A$: IF A
      $ =  CHR$ (13) AND NU = 1 THEN
       RETURN
 1046  IF A$ =  CHR$ (13) THEN NU =
      NU - 1: GOTO 1060
 1047 A =  VAL (A$): IF A < 1 OR A
       > 6 THEN  PRINT : GOTO 1040
 1050  VTAB 5 + (2 * A): HTAB 7: INV
      ERSE : PRINT A: NORMAL :DA$
```

```
 (NU) = HE$(A): IF NU > 5 THEN 1060
1057 NU = NU + 1: GOTO 1040
1060  HOME : PRINT "THE HEADINGS,
      IN ORDER, ARE:": VTAB 5: FOR
     I = 1 TO NU: PRINT DA$(I): PRINT
     : NEXT
1080  VTAB 21: PRINT "IS THIS COR
     RECT (Y/N) ?";: GET A$: IF A
     $ =  CHR$ (13) THEN  RETURN

1090  IF A$ <  > "Y" THEN 1000
1100  HOME : PRINT "ENTER THE LEN
     GTH OF EACH FIELD:": FOR I =
     1 TO NU
1130  VTAB 3 + (I * 2): PRINT I".
      "DA$(I);: HTAB 25: INPUT ""
     ;L$(I): IF  LEN (L$(I)) = 0 THEN
     1130
1135 L(I) =  VAL (L$(I)): IF L(I)
      = 0 OR L(I) > 65 THEN 1130
1140 L = L + L(I): PRINT : NEXT
1150  VTAB 21: PRINT "IS THIS COR
     RECT (Y/N) ?";: GET A$: IF A
     $ =  CHR$ (13) THEN  RETURN

1165  IF A$ <  > "Y" THEN 1100
1180  HOME : VTAB 5: INPUT "WHAT
     IS THE NAME OF THE MAGAZINE
     ?";MN$: IF  LEN (MN$) = 0 THEN
      RETURN
1200 RD = 0
1210  PRINT D$"OPEN HEADER."MN$
1220  PRINT D$"DELETE HEADER."MN$

1230  PRINT D$"OPEN HEADER."MN$
1235  PRINT D$"WRITE HEADER."MN$
1240  PRINT L: PRINT NU
1245  FOR I = 1 TO NU: PRINT DA$(
     I): PRINT L(I): NEXT
1250  PRINT D$"CLOSE HEADER."MN$
1255  PRINT D$"LOCK HEADER."MN$
1260  PRINT D$"OPEN DATA."MN$",L"
     L
1270  PRINT D$"WRITE DATA."MN$",R
     O"
1280  PRINT RD
1290  PRINT D$"CLOSE": RETURN
2000  HOME :A$ = "LOAD A MAGAZINE
     ": GOSUB 60000: VTAB 5
2010  PRINT : PRINT "ENTER THE NA
     ME OF THE MAGAZINE"
2015  INPUT A$: IF A$ = "*" THEN
      PRINT D$"CATALOG": GOTO 201
     0
2020  IF  LEN (A$) = 0 THEN  RETURN

2025 MN$ = A$
2030  PRINT D$"OPEN HEADER."MN$
2040  PRINT D$"READ HEADER."MN$
2050  INPUT L: INPUT NU
2060  FOR I = 1 TO NU: INPUT DA$(
     I): INPUT L(I): NEXT
2070  PRINT D$"CLOSE HEADER."MN$
2080  PRINT D$"OPEN DATA."MN$",L"
     L
2090  PRINT D$"READ DATA."MN$",RO
     "
2100  INPUT RD
2110  PRINT D$"CLOSE"
2120  RETURN
3000  GOSUB 200: HOME :A$ = "ADD
     A RECORD": GOSUB 60000
3010  VTAB 5: PRINT "FOR EACH REC
     ORD, ENTER THE INFORMATION":
      PRINT "FOLLOWED BY A RETURN
         IF YOU MAKE A": PRINT "MI
     STAKE, PRESS THE <-- KEY TO
     REASE THE": PRINT "PREVIOUS
     LETTER.  IF YOU WANT TO FINI
     SH"
3020  PRINT "ADDING RECORDS, PRES
     S ";: INVERSE : PRINT "RETUR
     N";: NORMAL : PRINT " WHEN A
     SKED FOR ";DA$(1)"."
3025  PRINT : PRINT "IF AT ANY TI
     ME YOU WISH TO ABORT THE": PRINT
     "RECORD, PRESS CTRL-R TO RET
     URN TO MENU."
3030  VTAB 18:A$ = "PRESS ANY KEY
      TO BEGIN": GOSUB 60000
3040 X =  PEEK ( - 16384): IF X <
     128 THEN 3040
3055 RD = RD + 1: IF RD > 500 THEN
      HOME : VTAB 11: PRINT "YOU
     CAN'T ENTER ANTMORE RECORDS.
      IF YOU": PRINT "WANT TO ADD
     MORE, CREATE ANOTHER FILE."
     : FOR I = 1 TO 3000: NEXT : RETURN
3060  POKE  - 16368,0: HOME : PRINT
     "RECORD #"RD: VTAB 3
3070  FOR I = 1 TO NU:M =  LEN (D
     A$(I)): IF M > K THEN K = M +
     1
3080  NEXT
```

```
3085  FOR X = 1 TO NU: PRINT DA$(
     X);: HTAB K: PRINT ":";: FOR
     I = 1 TO L(X): PRINT  CHR$ (
     95);: NEXT
3087  IF L(X) + K + 1 > 40 THEN  VTAB
      PEEK (37)
3090  HTAB K + 1:BE = 0:BD = 1: GOSUB
     3200: PRINT : PRINT
3105  IF ( LEN (BB$) = 0 AND X =
     1) OR B$ =  CHR$ (18) THEN R
     D = RD - 1: GOTO 100
3110 DI$(X) = BB$:BB$ = "": NEXT
3120  VTAB 19: HTAB 1: PRINT "IS
     THIS CORRECT (Y/N) ?";: GET
     A$: PRINT : IF A$ =  CHR$ (1
     3) THEN RD = RD - 1: RETURN
3125  IF A$ <  > "Y" THEN 3060
3151  PRINT D$"OPEN DATA."MN$",L"
     L
3160  PRINT D$"WRITE DATA."MN$",R
     "RD
3170  FOR I = 1 TO NU: PRINT  CHR$
     (34);DI$(I): NEXT
3180  PRINT D$"CLOSE"
3195  GOSUB 1260: GOTO 3055
3200  REM   ** INPUT ROUTINE **
3210  GET B$: IF B$ =  CHR$ (13) OR
     B$ =  CHR$ (18) THEN  RETURN
3220  IF B$ =  CHR$ (8) THEN 3260
3230  IF B$ <  CHR$ (32) THEN 321
     0
3232  IF B$ =  CHR$ (34) THEN 321
     0
3233  IF BE = 1 THEN  PRINT  CHR$
     (7);: GOTO 3210
3234 BD = 0
3235  IF  LEN (BB$) = L(X) THEN  PRINT
      CHR$ (7);:BE = 1: GOTO 3210
3240 BB$ = BB$ + B$: PRINT B$;:B$
     = "": GOTO 3210
3260  IF BD = 1 THEN  PRINT  CHR$
     (7);: HTAB K + 1: GOTO 3210
3265  IF  LEN (BB$) < 2 THEN BB$ =
     "":BD = 1: GOTO 3280
3270 BB$ =  LEFT$ (BB$, LEN (BB$)
      - 1)
3280 BE = 0: PRINT  CHR$ (8); CHR$
     (95); CHR$ (8);: GOTO 3210
4000  GOSUB 200: HOME :A$ = "CHAN
     GE AN ARTICLE": GOSUB 60000:
      VTAB 5
4006  IF RD = 0 THEN  PRINT "THER
     E ARE NO RECORDS.": FOR I =
     1 TO 2000 -  NEXT : RETURN
4010  INPUT "RECORD # TO CHANGE (
     -99 FOR ALL) ?";M1$: IF  LEN
     (M1$) = 0 OR  VAL (M1$) = 0 THEN
      RETURN
4025 M1 =  VAL (M1$): IF M1 > RD THEN
     4000
4030  IF M1 < 0 THEN M2 = 1:M3 =
     RD: GOTO 4050
4040 M2 = M1:M3 = M2
4045  VTAB 9
4050  VTAB 9: PRINT : PRINT "AFTE
     R EACH RECORD, I WILL ASK IF
      YOU": PRINT "WANT IT CHANGE
     D. IF NOT, I WILL READ IN": PRINT
     "THE NEXT RECORD. IF YOU DO
     WANT IT": PRINT "CHANGED, TH
     EN THE CURSOR WILL BE DIS-"
4060  PRINT "PLAYED AT THE BEGINN
     ING OF THE DATA. IF": PRINT
     "YOU WISH TO SKIP THE DATA,
     PRESS RETURN": PRINT "TO GO
     ON TO THE NEXT LINE OR CTRL-
     R FOR": PRINT "NEXT RECORD.
     OTHERWISE, ENTER IN DATA."
4090  VTAB 21:A$ = "PRESS ANY KEY
      TO BEGIN": GOSUB 60000
4100 X =  PEEK ( - 16384): IF X <
     128 THEN 4100
4110  HOME : POKE  - 16368,0
4120  FOR E = M2 TO M3
4130  PRINT "RECORD #"E: VTAB 5: GOSUB
     6500: GOSUB 6300
4170  VTAB 3 + (X * 2) + 2: PRINT
     "CHANGE THIS RECORD ?";: GET
     A$: IF A$ =  CHR$ (13) THEN
     RETURN
4190  IF A$ <  > "Y" THEN  HOME :
     NEXT : RETURN
4205  PRINT
4210  FOR X = 1 TO NU: VTAB 3 + (
     2 * X): HTAB K + 1:BE = 0:BD
     = 0: GOSUB 3200
4220  IF B$ =  CHR$ (13) AND  LEN
     (BB$) = 0 THEN  NEXT : GOTO
     4250
```

```
4230  IF B$ =  CHR$ (18) THEN X =
     NU: NEXT : HOME : NEXT : RETURN
4240 DI$(X) = BB$:BB$ = "": NEXT
4250  VTAB 23: CALL  - 958: PRINT
     "IS THIS CORRECT (Y/N) ?";: GET
     A$: IF A$ =  CHR$ (13) THEN
      RETURN
4270  IF A$ <  > "Y" THEN  HOME :
      GOTO 4130
4290  PRINT
4300  PRINT D$"OPEN DATA."MN$",L"
     L
4310  PRINT D$"WRITE DATA."MN$",R
     "E
4320  FOR T = 1 TO NU: PRINT  CHR$
     (34);DI$(T): NEXT
4330  PRINT D$"CLOSE"
4340  HOME : NEXT : RETURN
5000  GOSUB 200: HOME :A$ = "DELE
     TE AN ARTICLE": GOSUB 60000:
      VTAB 5
5015  IF RD = 0 THEN  PRINT "THER
     E ARE NO RECORDS TO DELETE."
     : FOR I = 1 TO 2000: NEXT : RETURN
5020  PRINT "WHICH RECORD TO ";: FLASH
     : PRINT "DELETE";: NORMAL : INPUT
     " ";M3$: IF  LEN (M3$) = 0 THEN
      RETURN
5040 M3 =  VAL (M3$): IF M3 < 1 OR
     M3 > RD THEN 5000
5050  PRINT : PRINT "ARE YOU SURE
      (YES/N)";: INPUT A$: IF A$ <
      > "YES" THEN  RETURN
5060  IF M3 <  > RD THEN 5080
5070 RD = RD - 1: GOTO 1260
5080 E = RD: GOSUB 6500
5090  PRINT D$"OPEN DATA."MN$",L"
     L
5100  PRINT D$"WRITE DATA."MN$",R
     "M3
5110  FOR I = 1 TO MN: PRINT  CHR$
     (34);DI$(I): NEXT
5120  PRINT D$"CLOSE"
5130  GOTO 5070
6000  GOSUB 200: HOME :A$ = "LIST
      AN ARTICLE(S)": GOSUB 60000
     : GOSUB 6400
6010  IF PR = 1 THEN  PRINT D$"PR
     #1"
6020  FOR E = M1 TO M2: HOME : PRINT
     "RECORD #";E: GOSUB 6500: GOSUB
     6300
6030  PRINT
6045  PRINT D$"PR#0"
6050  VTAB 3 + (2 * X) + 2: INVERSE
     : PRINT "RETURN TO QUIT OR O
     THER TO CONTINUE";: NORMAL :
      GET A$: IF A$ =  CHR$ (13) THEN
     PR = 0: RETURN
6083  IF PR = 1 THEN  PRINT : PRINT
     D$"PR#1"
6085  NEXT
6087  PRINT : PRINT D$"PR#0"
6088  HOME : VTAB 11:A$ = "THAT'S
      ALL": GOSUB 60000: FOR I =
     1 TO 3000: NEXT :PR = 0: RETURN
6300  FOR X = 1 TO NU:M =  LEN (D
     A$(X)): IF M > K THEN K = M +
     1
6310  NEXT
6315  IF PR = 1 THEN 6330
6320  FOR X = 1 TO NU: VTAB 3 + (
     2 * X): PRINT DA$(X); TAB( K
     );":";DI$(X): PRINT : NEXT :
      RETURN
6330  PRINT : FOR X = 1 TO NU: PRINT
     DA$(X); TAB( K -  LEN (DA$(X
     )));":";DI$(X): NEXT : PRINT
     : PRINT : RETURN
6400  VTAB 5: PRINT "THERE ARE "R
     D" RECORDS."
6405  IF RD = 0 THEN  VTAB 5: PRINT
     "THERE AREN'T ANY RECORDS TO
      LIST.": FOR I = 1 TO 3000: NEXT
     : POP : RETURN
6410  VTAB 7: INPUT "WHAT IS THE
     STARTING RECORD TO LIST ?";M
     1$: IF  LEN (M1$) = 0 THEN  POP
     : RETURN
6420  IF  VAL (M1$) < 1 OR  VAL (
     M1$) > RD THEN 6410
6430 M1 =  VAL (M1$): VTAB 9: INPUT
     "WHAT IS THE ENDING RECORD T
     O LIST ?";M2$: IF  VAL (M2$)
      < M1 OR  VAL (M2$) > RD THEN
     6430
6440 M2 =  VAL (M2$): VTAB 11: PRINT
     "DO YOU WANT IT ON THE PRINT
     ER ?";: GET A$: PRINT A$: IF
     A$ <  > "Y" AND A$ <  > "N" THEN
     6440
```

```
6450  IF A$ = "Y" THEN  PRINT D$"
 PR#1":PR = 1
6460  RETURN
6500  PRINT D$"OPEN DATA."MN$",L"
 L
6510  PRINT D$"READ DATA."MN$",R"
 E
6520  FOR I = 1 TO NU: INPUT DI$(
 I): NEXT
6530  PRINT D$"CLOSE"
6540  RETURN
7000  HOME :A$ = "FIND/SORT OPTIO
 N": GOSUB 60000
7020  VTAB 5: PRINT "DO YOU WANT
 TO FIND OR SORT (F OR S) ?";
 : GET A$: IF A$ =  CHR$ (13)
  THEN  RETURN
7040  IF A$ = "S" THEN 7500
7050  IF A$ <  > "F" THEN 7000
7060  GOSUB 200: HOME :A$ = "FIND
 OPTION": GOSUB 60000
7070  VTAB 5: INPUT "WHICH STRING
  TO SEARCH FOR ?";ST$: IF  LEN
 (ST$) = 0 THEN  RETURN
7090  VTAB 7: FOR I = 1 TO NU: PRINT
 I"."DA$(I): PRINT : NEXT
7100  VTAB 23: INPUT "WHICH NUMBE
 R TO SEARCH ON ?";H$:H =  VAL
 (H$): IF H < 1 OR H > NU THEN
 7100
7125  HOME : GOSUB 6440: IF PR =
 1 THEN  PRINT D$"PR#1"
7130  PRINT : FOR E = 1 TO RD: HOME
 : GOSUB 6500:A$ = DI$(H)
7150  FOR I = 1 TO  LEN (A$) -  LEN
 (ST$) + 1
7160  IF  MID$ (A$,I, LEN (ST$)) =
 ST$ THEN  PRINT "RECORD #"E:
  GOSUB 6300: GOTO 7180
7170  NEXT : GOTO 7200
7180  PRINT D$"PR#0": VTAB 20: INVERSE
 : PRINT "RETURN TO EXIT OR O
 THER TO CONTINUE";: NORMAL :
  GET A$: PRINT : IF A$ =  CHR$
 (13) THEN PR = 0: RETURN
7190  IF PR = 1 THEN  PRINT : PRINT
 D$"PR#1"
7200  NEXT E:PR = 0: PRINT D$"PR#
 0"
7210  HOME : VTAB 12:A$ = "THAT'S
  ALL ''": GOSUB 60000: FOR I
  = 1 TO 2000: NEXT : RETURN
7280  NEXT E:PR = 0: PRINT D$"PR#
 0"
7500  GOSUB 200: HOME :A$ = "SORT
  OPTION": GOSUB 60000: VTAB
 4: IF RD < 2 THEN  PRINT "YO
 U CAN'T SORT LESS THAN 2 REC
 ORDS.": FOR I = 1 TO 3000: NEXT
 : RETURN
7510  FOR I = 1 TO NU: PRINT I;".
 ";DA$(I): PRINT : NEXT
7520  VTAB 23: INPUT "WHICH HEADI
 NG TO SORT ON ?";H$: IF  LEN
 (H$) = 0 THEN  RETURN
7530 H =  VAL (H$): IF H < 1 OR H
  > NU THEN 7520
7535  DIM A$(NU,RD),N$(RD): POKE
  768,29
7540  HOME : VTAB 11: HTAB 15: FLASH
 : PRINT "READING DISK";G$: NORMAL

7550  PRINT D$"OPEN DATA."MN$",L"
 L
7560  FOR I = 1 TO RD
7570  PRINT D$"READ DATA."MN$",R"
 I
7580  FOR J = 1 TO H: INPUT N$(I)
 : NEXT :B$ =  STR$ (I)
7581  IF  LEN (B$) = 1 THEN N$(I)
 = N$(I) + "00": GOTO 7583
7582  IF  LEN (B$) = 2 THEN N$(I)
 = N$(I) + "0"
7583 N$(I) = N$(I) + B$
7590  NEXT
7600  PRINT D$"CLOSE"
7605  HOME : VTAB 11: HTAB 17: FLASH
 : PRINT "SORTING";G$: NORMAL

7610  FOR I = 1 TO RD - 1: FOR J =
 I + 1 TO RD: IF N$(I) <  = N
 $(J) THEN 7630
7620 T$ = N$(I):N$(I) = N$(J):N$(
 J) = T$
7630  NEXT J,I:Q = 2
7640  HOME : VTAB 6: HTAB 14: FLASH
 : PRINT "SORT COMPLETED";G$;
 G$;G$: NORMAL
7650  VTAB 9: PRINT "DO YOU WANT
 TO SEE THE SORTED LIST ?";: GET
 A$: PRINT A$: IF A$ <  > "Y"
  THEN 7700
7660  GOSUB 6440: FOR Q = 1 TO RD
 : HOME : PRINT "RECORD #"Q:E
```

```
 =  VAL ( RIGHT$ (N$(Q),3)):
 GOSUB 6500: FOR J = 1 TO NU
:A$(J,Q) = DI$(J): NEXT : GOSUB
6300: PRINT D$"PR#0"
7670  VTAB 3 + (2 * X) + 2: INVERSE
: PRINT "RETURN TO QUIT OR O
THER TO CONTINUE";: NORMAL :
 GET A$: PRINT : IF A$ =  CHR$
(13) THEN PR = 0:Q = Q + 1: GOTO
7700
7680  IF PR = 1 THEN  PRINT D$"PR
#1"
7690  NEXT
7700  HOME : VTAB 5: PRINT D$;"PR
#0": PRINT "DO YOU WANT TO S
AVE THE SORTED FILE ?";: GET
A$: PRINT A$: IF A$ <  > "Y"
 THEN  RUN
7705  HOME : VTAB 11: HTAB 15: FLASH
: PRINT "WRITING FILE";G$: NORMAL

7710  PRINT D$"OPEN DATA."MN$",L"
L: FOR F = Q - 1 TO RD: PRINT
D$"READ DATA."MN$",R" VAL
( RIGHT$ (N$(F),3)): FOR J = 1
 TO NU: INPUT
A$(J,F): NEXT : NEXT
7715  PRINT D$"CLOSE": PRINT D$"O
PEN DATA."MN$",L"L
7720  FOR I = 1 TO RD: PRINT D$"W
RITE DATA."MN$",R"I
7730  FOR J = 1 TO NU: PRINT  CHR$
(34);A$(J,I): NEXT : NEXT
7740  PRINT D$"CLOSE": RUN
8000 ERR =  PEEK (222): IF ERR =
255 THEN  RESUME
8005  HOME : PRINT D$"PR#0": NORMAL
 : VTAB 5: PRINT G$
8010  IF ERR = 4 THEN  PRINT "DIS
K IS WRITE-PROTECTED.": PRINT
"PLACE ANOTHER IN DISK DRIVE
": GOTO 8200
8020  IF ERR = 5 THEN  PRINT "END
 OF DATA.": PRINT "I BET YOU
ENTERED A WRONG FILENAME!!!
": PRINT D$"DELETE HEADER."M
N$:MN$ = "": GOTO 8200
8030  IF ERR = 6 THEN  PRINT "FIL
E NOT FOUND!! YOU MUST HAVE
ENTERED THE WRONG FILENAME!!
": GOTO 8200
8040  IF ERR = 8 THEN  PRINT "I/O
 ERROR. CHECK DRIVE DOOR.": GOTO
8200
8050  IF ERR = 9 THEN  PRINT "DIS
K FULL. PLACE ANOTHER IN DIS
K DRIVE." GOTO 8200
8060  IF ERR = 10 THEN  PRINT "YO
U ALREADY HAVE A FILE BY THA
T NAME.": GOTO 8200
8070 LI =  PEEK (218) +  PEEK (21
9) * 256: PRINT "ERROR #"ERR
" IN LINE #"LI"." PRINT "CHE
CK APPLESOFT MANUAL, PAGE 13
6.": END
8200  PRINT : PRINT "PRESS ANY KE
Y FOR MAIN MENU.";: GET A$: GOTO
100
59999  HOME : POP : POKE 768,2: END
60000  PRINT  TAB( 28 -  LEN (A$)
 / 2);A$: RETURN
```

巧破密令答案揭曉

巧破密令有獎遊戲的舉辦，獲得讀者熱烈的支持，直到本月初截稿時，經已收到超過四百封的參加信，其中答中的超過百分之九十五。部份讀者可能一時心急，將答案寫錯，在此我們感到十分之抱歉。我們將會在參加信中抽出得獎者。然後以電話通知。得獎名單將於第四期電腦時代刊出，所有獎品將由第五期送出。

上期刊出之巧破密令有獎遊戲的答案是：

WE WISH YOU A MERRY CHRISTMAS
AND YOU CAN WIN THE PRIZE

動量不滅定律模擬法

□THOMAS

各位讀者在中學時也曾讀過這個題目，以下所介紹的程式就是對一個物體在無阻力下相碰撞後的情形作一探討。在運行程式前先讓我們重溫一下動量（MOMENTUM）的定義。當一個物體在碰撞後或發生爆炸後，其所產生的能量就是動量（動量就是一個物體的重量與它的速度相乘的數值）。當兩個沒有彈性的物體，如木頭車，在一個水平面碰撞時，靜止中的那一個與移動的那一個就會結合成一體而朝一個方向前進，而其所產生的能量，就在碰撞時化為聲音，光和熱，最終時就停止移動。

例一：（圖一）

假設一架重量為 60 KG 的大木頭車在以 3 公尺／秒的速度撞向一架重量 30 KG 的靜止木頭車，在動量不變定律下，我們就可以求出靜止的木頭車的速度為 2 公尺／秒。

但當兩個有彈性的物體，如球體相互碰撞時，其總動量仍保持不變，而兩個球體相撞後則朝兩個相反的向向反彈出去，其速度自然有了改變。

這個程式主要是模擬兩架沒有阻力（磨擦力）的木頭車在一個平滑的表面相碰撞，其速度及重量可以任意改變，但有一限制，就是它們的速度一定要少於 30 公分／秒，而重量則不可以少於零。當程式執行時大家會在畫面看到兩架車及一條路軌，而這路軌則是沒有空氣阻力的。這條路軌兩端皆有一屏障用來作為物體的反彈牆，路軌全長 106.6公分，而每架車之長度為 6.6公分。車的位置可以任意放在路軌的任何一點，而第一架車的位置則一定要在第二架車的左方，假如數據輸入錯誤，程式會要求再度輸入。數據輸入完畢後，程式便會開始描繪兩架車子及路軌的圖形，當車子在碰到反彈牆時便會被彈開，繼而與相反方向的車子相碰撞，這時彼此的重量便會產生一個總動量，再由此計算出不同車子在相撞後所得回之應有速度，由於是沒有能量的損耗（沒有阻力），所以兩架車子應該是永遠在路軌上移動。

假如想中途停止或輸入新的數據，可按< SPACE >鍵，按其他鍵則使程式繼續執行。□

```
1   REM  ***************************

2   REM  *   AIR CAR SIMULATION  *

3   REM  *  EDITED BY THOMAS LUK *

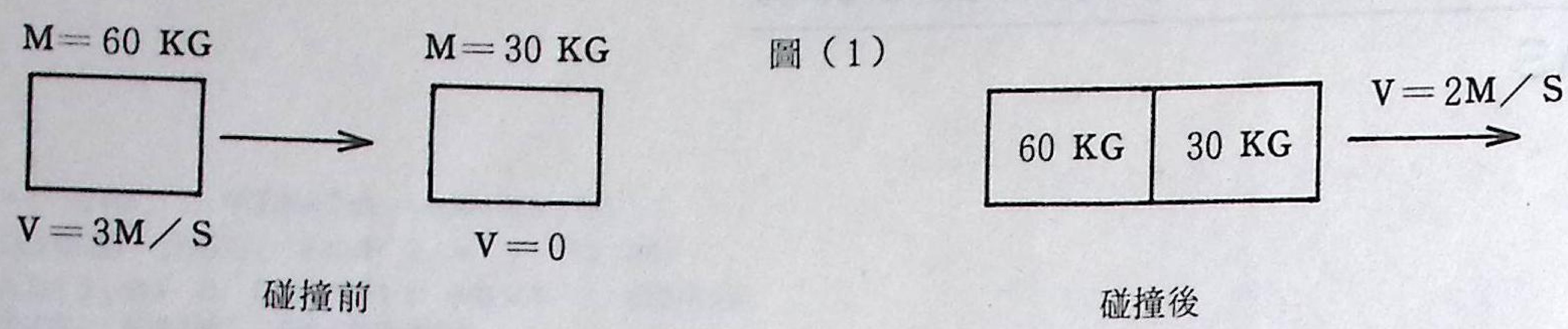
4   REM  * (C) MICRO ORCHID 1983 *

5   REM  ***************************

6   ONERR  GOTO 20
7   GOSUB 1000
10   GOSUB 540
20   PRINT : INPUT "PLEASE ENTER V
     1, M1, V2, M2 ";S1,M1,S2,M2
21   INVERSE
22   IF  ABS (S1) > 30 THEN  PRINT
     "V1 IS TOO LARGE, MUST BE LE
     SS THAN 30": NORMAL : GOTO 2
     0
23   IF  ABS (S2) > 30 THEN  PRINT
     "V2 IS TOO LARGE, MUST BE LE
     SS THAN 30": NORMAL : GOTO 2
     0
24   IF M1 < = 0 THEN  PRINT "MAS
     S MUST BE POSITIVE": NORMAL
     : GOTO 20
25   IF M2 < = 0 THEN  PRINT "MAS
     S MUST BE POSITIVE": NORMAL
     : GOTO 20
26   IF M1 / M2 > 100 THEN  PRINT
     "MAKE MASSES MORE EVEN": NORMAL
     : GOTO 20
27   IF M2 / M1 > 100 THEN  PRINT
     "MAKE MASSES MORE EVEN": NORMAL
     : GOTO 20
28 S1 = S1 / 3:S2 = S2 / 3
30 C1 = 3
40 C2 = 3
50 CO = 0
60 MR = (M1 - M2) / (M1 + M2)
70 F1 = (2 * M2) / (M1 + M2)
80 F2 = (2 * M1) / (M1 + M2)
90 SI = 16
100  HGR
110 Y = 140:XM = 10:XL = 254
120  NORMAL : INPUT "PLEASE ENTER
      X1 AND X2 ";X3,X4: INVERSE

121 X3 = X3 * 2.44:X4 = X4 * 2.44

122  IF X3 < 0 THEN  PRINT "CARS
     MUST BE ON THE TRACK": GOTO
     120
123  IF X3 > X4 - SI THEN  PRINT
     "FIRST CAR MUST BE ON THE LE
     FT": GOTO 120
124  IF X4 > 270 - SI THEN  PRINT
     "CARS MUST BE ON THE TRACK":
     GOTO 120
125 X3 = X3 + 10
126 X4 = X4 + 10
129  NORMAL
170  ROT= 0
```



圖（1）

```
180  SCALE= 1
190  HCOLOR= 1
200  HPLOT XM - 1,Y + 1 TO XL + S
     I - 1,Y + 1
210  HPLOT XM - 1,Y + 1 TO XM - 1
     ,Y - 5
220  HPLOT XL + SI - 1,Y + 1 TO X
     L + SI - 1,Y - 5
230 X1 = X3:X2 = X4
240  GOSUB 350
250  GOSUB 400
260 X1 = X3
270 X3 = X3 + S1
280 X2 = X4
290 X4 = X4 + S2
300  IF X3 > = X4 - SI AND S1 -
     S2 > 0 THEN  GOTO 450
310  IF X3 < = XM THEN S1 =  - (
     S1):X3 = XM
320  IF X4 > = XL THEN S2 =  - (
     S2):X4 = XL - 1
322  IF X4 < = XM + SI THEN S2 =
      - (S2):X4 = XM + SI + 1
323  IF X3 > = XL - SI THEN S1 =
      - (S1):X3 = XL - SI - 1
330 X =  PEEK (49152): POKE 49168
     ,0: IF X < > 160 THEN 240
335  PRINT : PRINT "TYPE C TO CON
     TINUE WITH OLD PARAMETERS ";
     : GET W$: IF W$ = "C" THEN 2
     40
337  PRINT : PRINT "TYPE L TO LEA
     VE THIS PROGRAM ";: GET W$: IF
     W$ < >"L" THEN 340
340  GOTO 20
350  HCOLOR= CO
360  DRAW 1 AT X1,Y
370  HCOLOR= C1
380  DRAW 1 AT X3,Y
390  RETURN
400  HCOLOR= CO
410  DRAW 1 AT X2,Y
420  HCOLOR= C2
430  DRAW 1 AT X4,Y
440  RETURN
450  IF  ABS (S1) <  ABS (S2) THEN
     X4 = X3 + SI
460  IF  ABS (S2) <  ABS (S1) THEN
     X3 = X4 - SI
470 T1 = S1
480 S1 = F1 * S2 + MR * S1
490 S2 = F2 * T1 - MR * S2
500  GOTO 310
510  GOTO 450
520  TEXT
530  STOP
540 AD = 32000
550  POKE 232, INT (AD -  INT (AD
     / 256) * 256)
560  POKE 233, INT (AD / 256)
570  READ D1
580  IF D1 > 255 THEN  RETURN
590  POKE AD,D1
600 AD = AD + 1
610  GOTO 570
620  DATA 1,255,4,0
630  DATA 45,45,45,45,45
640  DATA 45,45,36,36,36,63,63,63

650  DATA 63,63,63,63,54,54,54,00
     ,1000
1000  HOME
1005  PRINT
1010  HTAB (11)
1020  INVERSE : PRINT "AIR TRACK
     SIMULATION"
1025  NORMAL
1030  VTAB (5)
1040  HTAB (18)
1050  PRINT "FROM"
1060  PRINT : PRINT : PRINT
1070  HTAB 15: PRINT "MICRO ORCHI
     D"
1080  PRINT
1090  HTAB (19)
1100  PRINT
1110  PRINT
1120  HTAB 12: PRINT "COPYRIGHT (
     C) 1983"
1130  FOR I = 1 TO 2000: NEXT
1140  VTAB (23): PRINT "DO YOU NE
     ED INSTRUCTIONS?";: GET W$: IF
     W$ = "Y" THEN  GOSUB 1200
1150  CALL - 936
1160  VTAB (23)
1170  RETURN
1200  CALL - 936
1210  PRINT "  THIS PROGRAM SIMUL
     ATES AN AIR TRACK"
1220  PRINT : PRINT "WITH TWO FRI
     CTIONLESS AIR CARS. THESE"
1230  PRINT : PRINT "CARS HAVE MA
     SSES AND VELOCITIES THAT"
1240  PRINT : PRINT "YOU ENTER WH
     EN ASKED TO. I WILL DRAW A"
1250  PRINT : PRINT "TRACK AND MO
     VE THE TWO CARS ALONG THIS"
1260  PRINT : PRINT "TRACK THE WA
     Y TWO FRICTIONLESS CARS"
1270  PRINT : PRINT "WOULD MOVE A
```

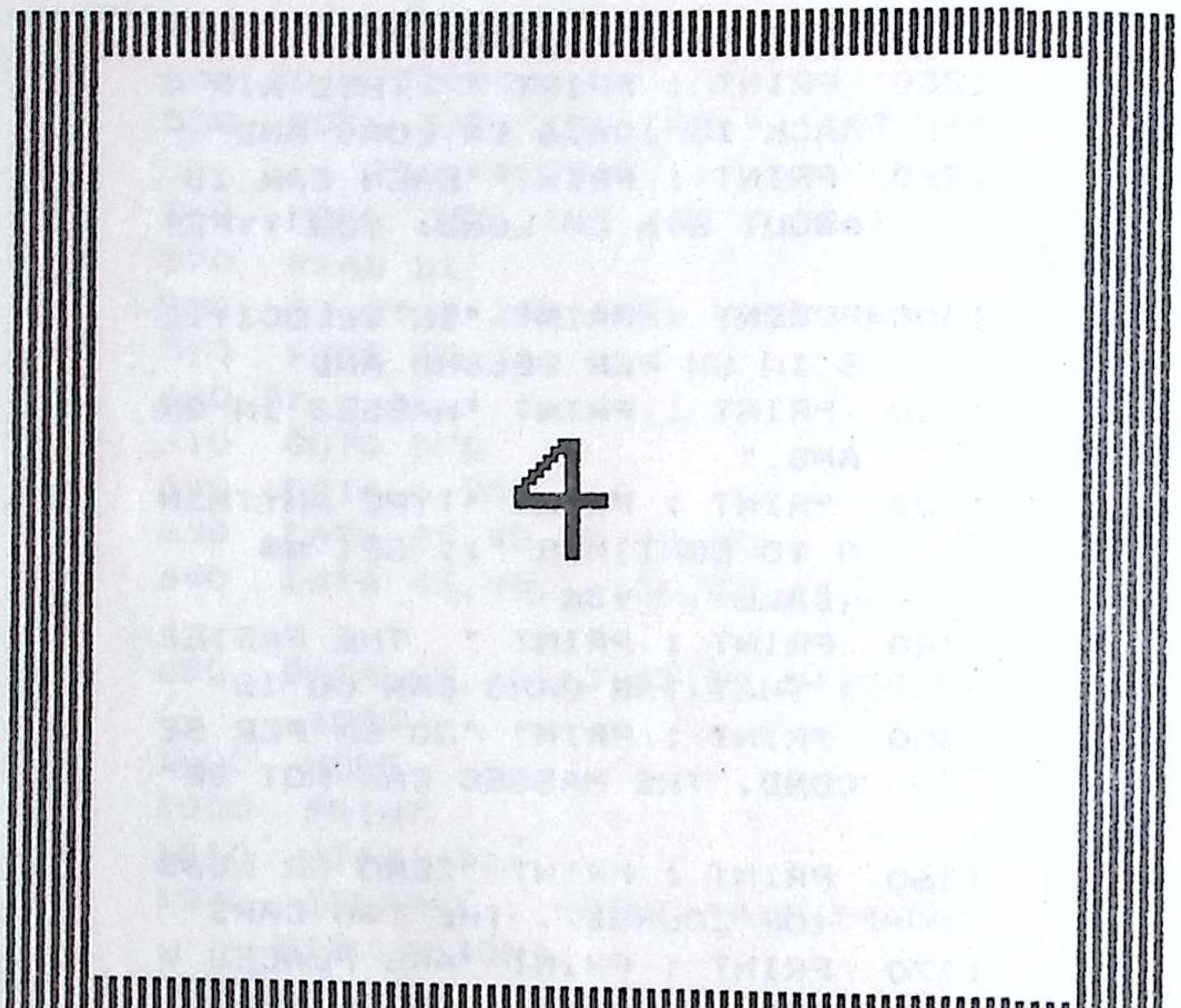
```
     LONG AN AIR TRACK."
1280  PRINT : PRINT "  THIS AIR T
     RACK IS 106.6 CM LONG AND"
1290  PRINT : PRINT "EACH CAR IS
     ABOUT 6.6 CM LONG. YOU TYPE"
1300  PRINT : PRINT "IN VELOCITIE
     S IN CM PER SECOND AND"
1310  PRINT : PRINT "MASSES IN GR
     AMS."
1320  PRINT : PRINT "TYPE ANYTHIN
     G TO CONTINUE ";: GET W$
1330  CALL  - 936
1340  PRINT : PRINT "  THE FASTES
     T THAT THE CARS CAN GO IS"
1350  PRINT : PRINT "30 CM PER SE
     COND. THE MASSES CAN NOT BE"
1360  PRINT : PRINT "ZERO OR LESS
      (OF COURSE). THE TWO CARS"
1370  PRINT : PRINT "ARE PLACED W
     HERE YOU WANT THEM ON THE"
1380  PRINT : PRINT "TRACK. THE F
     IRST CAR MUST BE PLACED TO"
1390  PRINT : PRINT "THE LEFT OF
     THE SECOND ONE. THE CARS"
1400  PRINT : PRINT "MUST BE PLAC
     ED ON THE TRACK. THEY MUST"
1410  PRINT : PRINT "BE PLACED BE
     TWEEN 0 AND 100. IF YOU"
1420  PRINT : PRINT "MAKE A MISTA
     KE, YOU WILL BE ASKED TO"
1430  PRINT : PRINT "RETYPE THE N
     UMBERS (SO TRY NOT TO MESS"
1440  PRINT : PRINT "UP)."
1460  PRINT : PRINT "TYPE ANYTHIN
     G TO CONTINUE ";: GET W$
1470  CALL  - 936
1480  PRINT : PRINT "  THE AIR CA
     RS BOUNCE OFF THE WALLS"
1490  PRINT : PRINT "WITHOUT ANY
     CHANGE IN THE MAGNITUDE OF"
1500  PRINT : PRINT "THEIR VELOCI
     TIES. HOWEVER, WHEN THE TWO"
1510  PRINT : PRINT "CARS COLLIDE
     , THE MASSES OF EACH CAR,"
1520  PRINT : PRINT "AS WELL AS T
     HE VELOCITIES OF THE CARS"
1530  PRINT : PRINT "COME INTO PL
     AY. ALL COLLISIONS IN THIS"
1540  PRINT : PRINT "PROGRAM ARE
     ASSUMED TO BE ELASTIC, THAT"
1550  PRINT : PRINT "IS TO SAY TH
     AT NO ENERGY IS LOST IN THE"
1560  PRINT : PRINT "COLLISIONS.
     THAT IS WHY THE CARS BOUNCE"
1570  PRINT : PRINT "OFF THE WALL
     S WITH THE SAME VELOCITY"
1580  PRINT : PRINT "THAT THEY HI
     T IT WITH."
1590  PRINT : PRINT "TYPE ANYTHIN
     G TO CONTINUE ";: GET W$
1600  CALL  - 936
1610  PRINT : PRINT "  THIS MEANS
     THAT WHEN THE TWO CARS"
1620  PRINT : PRINT "COLLIDE, THE
     TOTAL KINETIC ENERGY OF"
1630  PRINT : PRINT "THE SYSTEM M
     UST BE CONSERVED."
1640  PRINT : PRINT "THE EQUATION
     FOR KINETIC ENERGY IS :"
1650  PRINT : PRINT "   .5 * MASS
     * VELOCITY SQUARED"
1660  PRINT : PRINT "THE TOTAL EN
     ERGY OF THE SYSTEM WOULD BE"
1670  PRINT : PRINT "THE SUMS OF
     THE KINETIC ENERGIES OF THE"
1680  PRINT : PRINT "TWO CARS. TH
     IS MEANS THAT THE INDIVIDUAL
     "
1690  PRINT "KINETIC ENERGIES NEE
     D NOT STAY CONSTANT"
1700  PRINT : PRINT "BUT ONLY THA
     T THE TOTAL REMAIN CONSTANT.
     "
1710  PRINT : PRINT "TYPE ANYTHIN
     G TO CONTINUE";: GET W$
1720  CALL  - 936
1730  PRINT : PRINT "  THIS PROGR
     AMS SHOWS THE WAY CHANGES"
1740  PRINT : PRINT "IN THE PRE-C
     OLLISION VELOCITIES AND"
1750  PRINT : PRINT "RELATIVE MAS
     SES OF THE CARS EFFECT THE"
1760  PRINT : PRINT "POST-COLLISI
     ON VELOCITIES."
1770  PRINT : PRINT "  TO TRY NEW
     PARAMETERS, TYPE A SPACE"
1780  PRINT : PRINT "ANYTIME WHIL
     E THE CARS ARE IN MOTION."
1790  VTAB (23)
1800  PRINT "TYPE ANYTHING TO BEG
     IN GRAPHING ";: GET W$
1810  RETURN
```

適合稚齡兒童學習
幼童認數目字教育程

本文所介紹的程式，是供兒童學習數字之用。它可幫助兒童學習及認識 0 至10的數字，其的特點，是利用聲音作輔助，將數字生動地在螢幕上表達出來。筆者曾將此程式用於一個兩歲大的幼童，發覺效果頗爲理想。

當運行程式時，首先螢幕會出現簡介，解釋程式的內容，隨之是一個高解像隨機數字出現在螢幕上。這時用家應向兒童正確地讀出數字，並吩咐兒童立即依樣重讀一次。然後，兒童應從鍵盤上，找出類似螢幕所示之數字鍵掣。當監察兒童所做全無錯時，再次要求兒童在全無輔助情況下，讀出數字及找出同一的數字鍵掣。如果所按鍵掣正確的話，螢幕便出現一隻大家都喜愛的高解像老鼠圖畫及音樂。其實，簡單來說，程式的要點，是要兒童讀出螢幕顯示的數字。

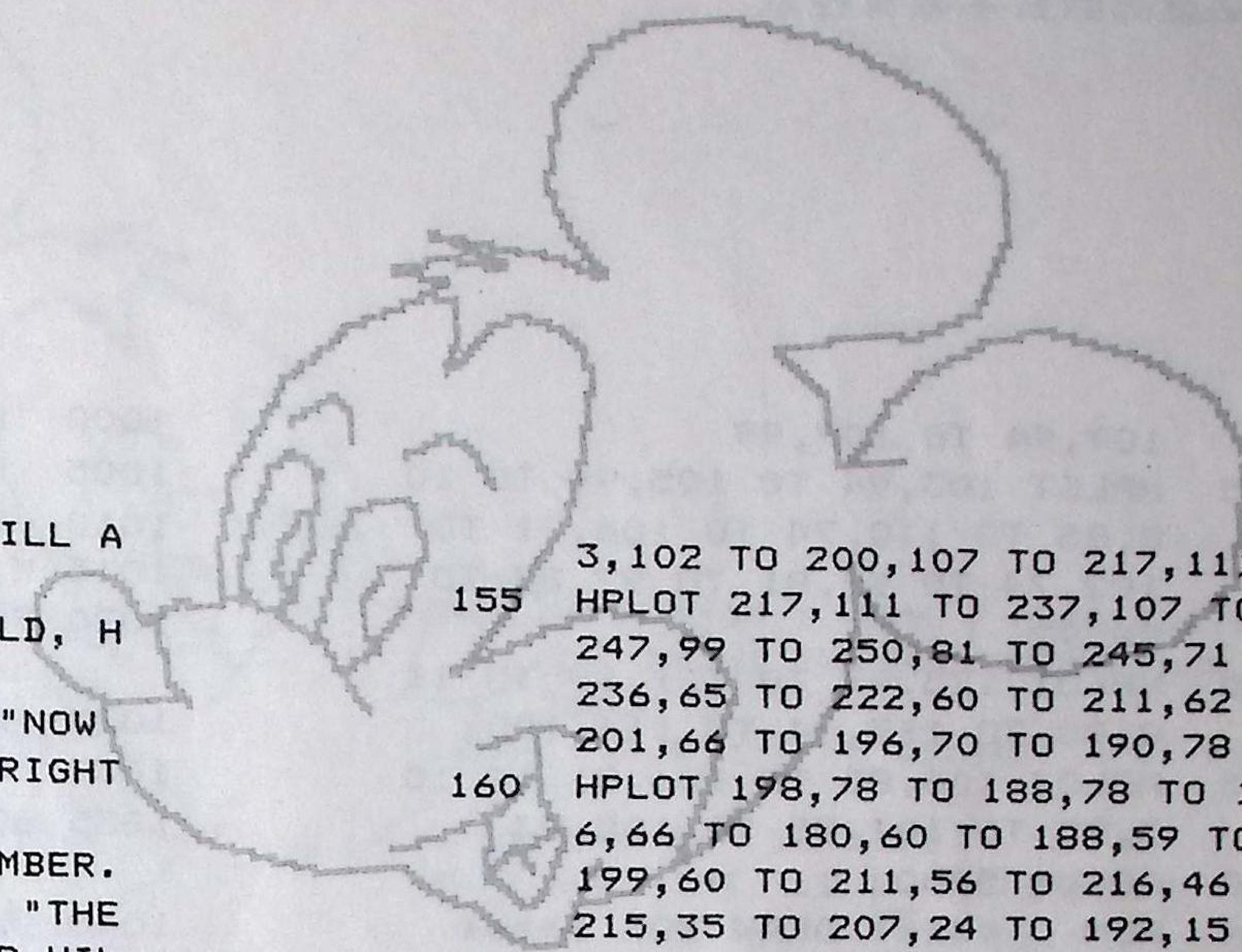
在輸入圖形表（SHAPE TABLE）後，應按入 BSAVE NUMBER TABLE，A$ 6000，L$ 01CD 及按 RETURN掣。然後，可輸入由 0302 至 $ 0326 聲音產生器及按入 BSAVE MSOUND，A$ 0302，L$0015 和按 RETURN掣。最後輸入 APPLESOFT 程式，存貯後已可準備開始。■



程式（1）幼童認數目字教育程式

```
]LIST

1  REM    **********************
    *
2  REM    *  BOUNCING NUMBERS
    *
3  REM      * BY COMPUHOUSE LTD
    *
4  REM    * COPYRIGHT (C) 1983
    *
7  REM    **********************
    *
10  TEXT : HOME : VTAB 12: PRINT
    "******** (C) MICRO ORCHID 1
    983 ********": PRINT  CHR$ (
    4);"BLOAD MSOUND"
15  PRINT  CHR$ (4);"BLOAD NUMBER
    TABLE": HIMEM: 8192: DIM A(
    11):A = 37: HOME : SPEED= 0:
    SN =  - 16336
20  FOR I = 3 TO 10: HTAB I: VTAB
    10
25  PRINT "*":SO =  PEEK (SN) -  PEEK
    (SN):A = A - 2: HTAB (I + A)
    : VTAB 10: PRINT "*":SO =  PEEK
    (SN) -  PEEK (SN)
30  NEXT I: SPEED= 255
35  VTAB 10: HTAB 12: PRINT "NUMB
    ER RECOGNITION"
40  FOR L = 1 TO 11: FOR T = 1 TO
    500: NEXT T: CALL  - 912
45 SO =  PEEK (SN) -  PEEK (SN): NEXT
    L
50  HOME : PRINT "THIS IS NUMBER
    RECOGHITION FOR"
55  PRINT "PRESCHOOLERS. THE IDEA
    OF THIS PROGRAM": PRINT "IS
    TO TEACH PRESCHOOLERS HOW T
    O"
60  PRINT "RECOGHIZE NUMBER SHAPE
    S."
```



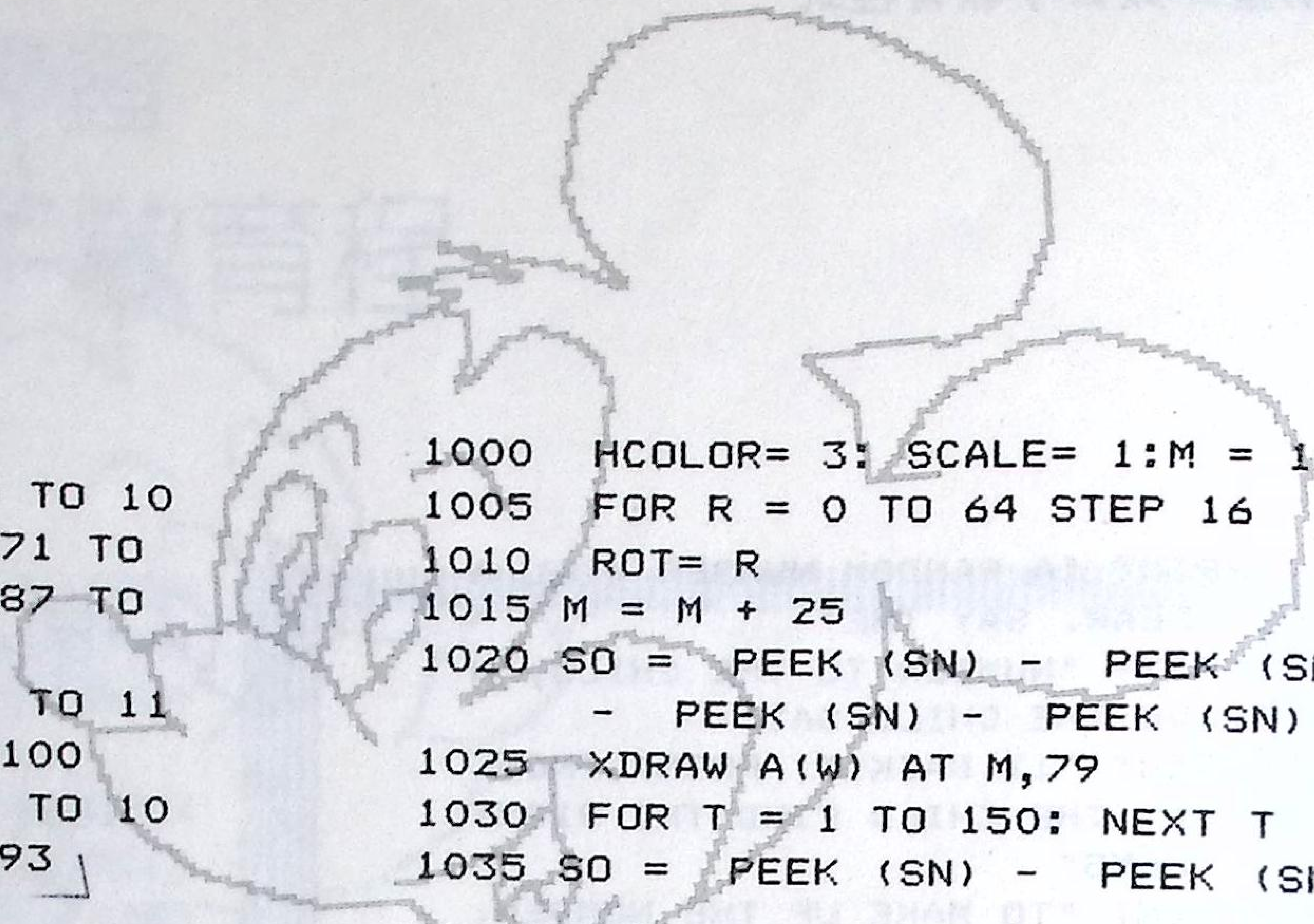
```
65   PRINT "A RANDOM NUMBER WILL A
      PPEAR. SAY THE"
70   PRINT "NUMBER TO THE CHILD, H
      AVE THE CHILD SAY"
75   PRINT "IT BACK.": PRINT "NOW
      LET THE CHILD FIND THE RIGHT
       KEYS"
80   PRINT "TO MAKE UP THE NUMBER.
       AFTER PRESSING": PRINT "THE
       RIGHT KEYS A NEW NUMNER WIL
      L COME UP"
85   VTAB 22: PRINT "*** HIT ANY K
      EY TO CONTINUE ***": GET A$
90   HOME : HTAB 8: VTAB 10: PRINT
      "TYPE IN THE (+) CHA RACTER"

95   HTAB 11: VTAB 12: PRINT "TO E
      ND THE PROGRAM"
100  VTAB 22: PRINT "*** HIT ANY
      KEY TO CONTINUE ***": GET A$

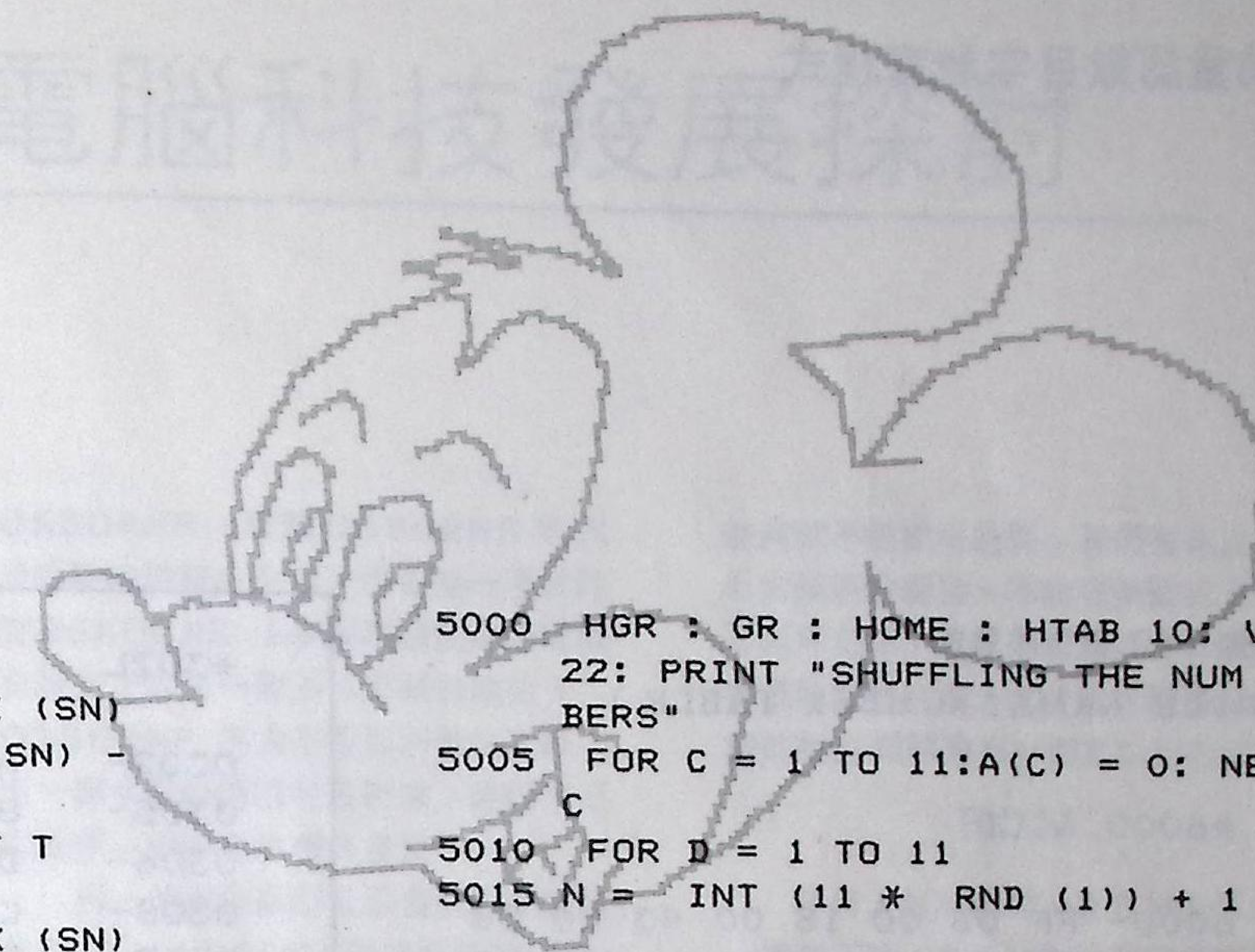
105  HOME : VTAB 12: PRINT "HOW M
      ANY GUESSES AT EACH NUMBER W
      OULD": PRINT "YOU LIKE."
110  PRINT : INPUT "INPUT NUMBER
      OF GUESSES (1 TO 5) AND    H
      IT RETURN      ";G: PRINT : PRINT

115  IF (G > 0) AND (G < 6) THEN
      125
120  PRINT "THAT WAS NOT BETWEEN
      1 AND 5": FOR T = 1 TO 2000:
       NEXT T: GOTO 105
125  HOME : HTAB 7: VTAB 11: PRINT
      "DO YOU WANT MUSIC (YES / NO
      )?": GET Z$
130  IF (Z$ = "N") OR (Z$ = "Y") THEN
      140
135  GOTO 125
139  REM  ***** DRAW MOUSE *****
140  POKE 232,1: POKE 233,96: HGR2
      : TEXT : HOME : HCOLOR= 3
145  HPLOT 75,106 TO 76,111 TO 79
      ,119 TO 90,128 TO 105,133 TO
      118,133 TO 124,138 TO 129,14
      0 TO 136,146 TO 142,145 TO 1
      48,134
150  HPLOT 145,140 TO 154,138 TO
      165,132 TO 182,118 TO 187,11
      0 TO 189,102 TO 189,93 TO 19
      3,102 TO 200,107 TO 217,111
155  HPLOT 217,111 TO 237,107 TO
      247,99 TO 250,81 TO 245,71 TO
      236,65 TO 222,60 TO 211,62 TO
      201,66 TO 196,70 TO 190,78
160  HPLOT 198,78 TO 188,78 TO 18
      6,66 TO 180,60 TO 188,59 TO
      199,60 TO 211,56 TO 216,46 TO
      215,35 TO 207,24 TO 192,15
165  HPLOT 192,15 TO 178,11 TO 16
      5,11 TO 155,15 TO 149,23 TO
      147,33 TO 150,41 TO 155,48 TO
      146,44 TO 129,40 TO 140,46 TO
      133,44
170  HPLOT 133,44 TO 123,46 TO 13
      1,48 TO 124,51 TO 115,54 TO
      105,61 TO 96,75 TO 94,84 TO
      94,93 TO 89,92 TO 82,95
175  HPLOT 82,95 TO 79,89 TO 76,8
      7 TO 70,87 TO 66,91 TO 65,98
       TO 70,103 TO 75,106 TO 88,1
      07 TO 85,103 TO 83,94
180  HPLOT 95,92 TO 102,95 TO 109
      ,100 TO 112,105 TO 115,110
185  HPLOT 117,130 TO 125,128 TO
      134,125 TO 140,120 TO 140,11
      5 TO 140,124 TO 143,130 TO 1
      41,135 TO 138,140 TO 136,139
       TO 131,135 TO 133,125
190  HPLOT 142,129 TO 137,132 TO
      138,135 TO 136,131 TO 132,13
      5
195  HPLOT 131,116 TO 137,114 TO
      144,115 TO 149,117
200  HPLOT 160,135 TO 166,122 TO
      170,111 TO 167,104 TO 161,99
       TO 153,97 TO 142,100 TO 128
      ,105 TO 138,95 TO 147,83 TO
      150,74 TO 152,64
205  HPLOT 152,64 TO 150,57 TO 14
      6,54 TO 143,53 TO 137,55 TO
      130,61 TO 132,56 TO 132,50 TO
      129,50
210  HPLOT 136,78 TO 133,72 TO 13
      0,70 TO 123,73
215  HPLOT 115,70 TO 114,64 TO 11
      3,63 TO 106,67
220  HPLOT 111,103 TO 117,100 TO
      122,94 TO 125,87 TO 123,79 TO
      120,79 TO 114,81 TO 111,87 TO
```

幼童認數目字教育程式



```
      109,94 TO 107,99
225  HPLOT 103,94 TO 105,90 TO 10
      8,85 TO 110,74 TO 106,71 TO
      102,74 TO 99,81 TO 97,87 TO
      98,92
230  HPLOT 123,89 TO 121,89 TO 11
      8,90 TO 115,94 TO 116,100
235  HPLOT 108,80 TO 106,81 TO 10
      3,85 TO 102,88 TO 105,93
240  GOSUB 5000
244  REM  ***** DRAW BOX *****
245 W = 0: HGR
250 W = W + 1:TRY = 0
255  IF W = 12 THEN 240
260  IF Q = 1 THEN 340
265 A =  INT (7 *  RND (1)) + 1
270  IF A = 4 THEN 265
275  FOR I = 10 TO 139 STEP 2
280  HCOLOR= A
285  HPLOT 50,I TO 60,I
290  NEXT
295  FOR I = 140 TO 145 STEP 2
300  HPLOT 50,I TO 220,I
305  NEXT
310  FOR I = 144 TO 10 STEP  - 2
315  HPLOT 220,I TO 230,I
320  NEXT
325  FOR I = 10 TO 15 STEP 2
330  HPLOT 220,I TO 60,I
335  NEXT
340 B =  INT (7 *  RND (1)) + 1
345  IF B = 4 THEN 340
350  FOR I = 9 TO 140 STEP 2
355  HCOLOR= B
360  HPLOT 50,I TO 60,I
365  NEXT
370  FOR I = 141 TO 145 STEP 2
375  HPLOT 50,I TO 220,I
380  NEXT
385  FOR I = 145 TO 9 STEP  - 2
390  HPLOT 220,I TO 230,I
395  NEXT
400  FOR I = 9 TO 14 STEP 2
405  HPLOT 220,I TO 60,I
410  NEXT
415 Q = 1
420 E =  INT (3 *  RND (1)) + 1
425  ON E GOTO 1000,2000,3000
999  REM  SPIN NUMBER ONTO SCREEN
1000  HCOLOR= 3: SCALE= 1:M = 10
1005  FOR R = 0 TO 64 STEP 16
1010  ROT= R
1015 M = M + 25
1020 SO =  PEEK (SN) -  PEEK (SN)
      -  PEEK (SN) -  PEEK (SN)
1025  XDRAW A(W) AT M,79
1030  FOR T = 1 TO 150: NEXT T
1035 SO =  PEEK (SN) -  PEEK (SN)
1040  XDRAW A(W) AT M,79
1045  NEXT R
1050 SO =  PEEK (SN) -  PEEK (SN)
      PEEK (SN)
1055  DRAW A(W) AT 139,79
1060  GOSUB 6000
1065  XDRAW A(W) AT 139,79: GOTO
      250
1999  REM  RUN NUMBER ONTO SCREEN
2000  HCOLOR= 3: ROT= 0: SCALE= 1
2005  FOR I = 265 TO 139 STEP  -
      15
2010 SO =  PEEK (SO) -  PEEK (SN)
2015  XDRAW A(W) AT I,79
2020  NEXT
2025  FOR I = 265 TO 139 STEP  -
      15
2030 SO =  PEEK (SO) -  PEEK (SO)
2035  XDRAW A(W) AT I,79
2040  NEXT
2045 SO =  PEEK (SN) -  PEEK (SN)
2050  DRAW A(W) AT 139,79
2055  GOSUB 6000
2060  XDRAW A(W) AT 139,79: GOTO
      250
2656 A =  INT (7 *  RND (1)) + 1
2999  REM  POP NUMBER ONTO SCREEN
3000  HCOLOR= 3: SCALE= 1: ROT= 0
3005  FOR I = 1 TO 5
3010 X =  INT (200 *  RND (1)) +
      20
3015 Y =  INT (100 *  RND (1)) +
```



```
   20
3020  XDRAW A(W) AT X,Y
3025 SO =  PEEK (SN) -  PEEK (SN)
   -  PEEK (SN) - PEEK (SN) -
   PEEK (SN) -  PEEK (SN)
3030  FOR T = 1 TO 150: NEXT T
3035  XDRAW A(W) AT X,Y
3040 SO =  PEEK (SN) -  PEEK (SN)
   -  PEEK (SN) -  PEEK (SN) -
   PEEK (SN) -  PEEK (SN)
3045  NEXT I
3050  DRAW A(W) AT 139,79
3055 SO =  PEEK (SN) -  PEEK (SN)

3060  GOSUB 6000
3065  XDRAW A(W) AT 139,79: GOTO
   250
3999  REM  SHOW HGR2 WITH OR WITH
   OUT MUSIC
4000  POKE  - 16302,0: POKE  - 16
   299,0
4005  IF Z$ = "N" THEN 4070
4010  POKE 768,193: POKE 769,150:
   CALL 770
4015  POKE 768,193: POKE 769,50: CALL
   770
4020  POKE 768,193: POKE 769,175:
   CALL 770
4025  POKE 768,193: POKE 769,150:
   CALL 770
4030  POKE 768,193: POKE 769,50: CALL
   770
4035  POKE 768,193: POKE 769,175:
   CALL 770
4040  POKE 768,153: POKE 769,100:
   CALL 770
4045  POKE 768,193: POKE 769,80: CALL
   770
4050  POKE 768,172: POKE 769,80: CALL
   770
4055  POKE 768,199: POKE 769,100:
   CALL 770
4060  POKE 768,193: POKE 769,100:
   CALL 770
4065  POKE  - 16302,0: POKE  - 16
   300,0: RETURN
4070  FOR I = 0 TO 2000: NEXT : POKE
   - 16302,0: POKE  - 16300,0:
   RETURN
4999  REM   * SHUFFLE NUMBERS *
5000  HGR : GR : HOME : HTAB 10: VTAB
   22: PRINT "SHUFFLING THE NUM
   BERS"
5005  FOR C = 1 TO 11:A(C) = 0: NEXT
   C
5010  FOR D = 1 TO 11
5015 N =  INT (11 *  RND (1)) + 1
5020  FOR F = 1 TO 11: IF A(F) =
   N THEN 5015
5025  COLOR= N:X =  INT (39 *  RND
   (1)):Y =  INT (39 *  RND (1)
   ): PLOT X,Y
5030  SO =  PEEK ( - 16336): NEXT
   F
5035  A(D) = N: NEXT D
5040  GR : HOME :Q = 0: RETURN
5999  REM  KEYBOARD INPUT ROUTINE
6000  IF A(W) = 11 THEN 6045
6005  GET A$:A =  ASC (A$): IF A =
   43 THEN  TEXT : HOME : VTAB
   12: HTAB 15: PRINT "SEE YOU
   SOON": END
6010  IF (A < 48) OR (A > 57) THEN
   6005
6015 A =  VAL (A$): IF A <  > A(W
   ) - 1 THEN 6030
6020  GOSUB 4000
6025  RETURN
6030  FOR I = 0 TO 100:SO =  PEEK
   ( - 16336): NEXT I
6035 TRY = TRY + 1: IF TRY = G THEN
   RETURN
6040  GOTO 6000
6045  GET A$:A =  ASC (A$): IF A =
   43 THEN  TEXT : HOME : VTAB
   12: HTAB 15: PRINT "SEE YOU
   SOON": END
6050  IF (A < 48) OR (A > 57) THEN
   6045
6055  GET B$:B =  ASC (B$): IF (B
   < 48) OR (B > 57) THEN 6055

6060 A =  VAL (A$):B =  VAL (B$):
   IF (A = 1) AND (B = 0) THEN
   6070
6065  GOTO 6030
6070  GOSUB 4000
6075  RETURN
```

幼童認數目字教育程式

程式（3）數字圖形表

（FILE NAME：NUMBER TABLE）

```
*6000.61CD

6000- FF 0B 00 18 00 43 00 58
6008- 00 7F 00 A6 00 CC 00 F9
6010- 00 20 01 42 01 72 01 97
6018- 01 09 2D 2E 2E 2C 2E 2C
6020- 2E 2C 0E 20 25 25 27 25
6028- 27 25 27 25 27 25 27 27
6030- 27 37 27 37 27 37 E7 32
6038- 37 37 35 37 35 37 35 37
6040- 35 37 2D 00 49 91 29 3C
6048- 2C 3C 2C 3C 2C 3C 2C 3C
6050- 2C 3C 2C 3C 2C 3C 37 07
6058- 00 36 2E 2D 2D 2D 2D 2D
6060- 25 3F 3F 3F 3F 3F 27 2C
6068- 2C 2C 25 2D 2C 25 2D 2C
6070- 3C 2C 3C 3C 3C 3C 3E 3C
6078- 3E 3C 3E 3C 3E 3E 06 00
6080- 2E 35 25 35 25 35 25 35
6088- 25 25 25 25 27 C5 27 3F
6090- 3F 3F 27 2D 2D 2D 25 25
6098- 25 27 C5 27 27 37 27 37
60A0- 27 37 27 37 3F 36 00 49
60A8- 49 53 25 27 25 27 25 27
60B0- 25 2D 2D 3C 3F 27 27 25
60B8- 27 25 27 25 3F 3E 3E 3E
60C0- 3E 3E 3E 36 25 35 25 35
60C8- 25 35 25 04 00 2C 36 35
60D0- 25 35 25 35 25 35 25 25
60D8- 25 25 27 C5 27 27 37 27
60E0- 37 27 37 27 37 3F 2C 3C
60E8- 2C 3C 2C 3C 2C 35 25 35
60F0- 25 35 25 35 25 35 25 35
60F8- 06 00 35 35 25 35 25 35
6100- 25 75 20 25 25 27 C5 27
6108- 27 37 27 37 27 37 27 37
6110- 3E 36 3E 24 24 25 08 25
6118- 25 25 2D 2C 25 2D 2C 05
6120- 00 49 52 2D 3C 2C 25 27
6128- 2D 3C 2C 25 27 2D 3C 2C
6130- 25 27 2D 24 37 27 37 27
6138- 37 27 37 27 37 27 37 27
6140- 37 27 00 01 38 2C 3C 2C
6148- 2C 24 3C 3C 2C 2C 2C 2C
6150- 2E 2C 2E 2C 2E 8C 2E 2E
6158- 3E 3E 3E 3F 3F 3F 2E 2D
6160- 2D 2D 2E 2E 3E 2E 3E 3E
6168- 3E 3E 3C 3E 3C 3E 3C 1E
6170- 20 3F 00 52 29 2C 2C 2C
6178- 2C 2C 24 3C 3E 3C 3E 3C
6180- 3C 3C 2C 3C 08 25 25 35
6188- 25 35 25 35 65 32 35 35
6190- 37 D5 37 3F 2E 1A 07 00
6198- 09 2C 2E 2C 2C 2C 3C 2C
61AQ- 3C 2C 3C 2C 3C 2C 3C 3C
61A8- 3C 3C 3E 9C 3E 3E 2E 3E
61B0- 2E 3E 2E 3E 2E 3E 2E 2E
61B8- F6 DB 3B 2C 3C 2C 3C 2C
61C0- 3C 2C 3C 2C 3C 2C 3C 2C
61C8- 24 3F 57 2D 00 00
*_
```

```
*302L

0302-   AD 30 C0    LDA   $C030
0305-   88          DEY
0306-   D0 05       BNE   $030D
0308-   CE 01 03    DEC   $0301
030B-   F0 09       BEQ   $0316
030D-   CA          DEX
030E-   D0 F5       BNE   $0305
0310-   AE 00 03    LDX   $0300
0313-   4C 02 03    JMP   $0302
0316-   60          RTS
0317-   00          BRK
0318-   00          BRK
0319-   00          BRK
031A-   00          BRK
031B-   00          BRK
031C-   00          BRK
031D-   00          BRK
031E-   00          BRK
031F-   00          BRK
0320-   00          BRK
*
```

程式（2）聲音產生器程式

（FILE NAME：SOUND GENERTOR MSOUND）

未來五年電腦科技發展探討

□ 林耀 □

1954年，美國依利諾州大學的體育館，安裝有當時世界第一部最先進的電腦系統。體育館特別裝上全球最大的一套空氣調節系統，以調節電腦操作時產生的巨量熱能。

這部ILIAC電腦是利用眞空管製造。爲避免在操作時其中一枝眞空管損壞而影響計算結果，這部電腦將每項計算進行三次，然後取其多數值。以當時來說，這部電腦可算了不起。曾幾何時，舊時的英雄到今日已談不上任何的光釆。三十年後的今日，一部TI－59可編程式計數機功能就比 ILIAC 更強大。這項成果是經過三十年來不斷的努力研究方可達致。不過，科技發展是一日千里，假如沒有核子戰爭的發生，未來十年人類在科學上的成就發展勢可比得上過往三十年所累積的。

到公元二千年，在電腦科技發達的西方國家，究竟會發展到那個地步呢？可以預言，屆時，如果人們遇有難題，電腦可幫助他解決及找出答案。微型電腦將會成爲大型電腦和普通市民互相溝通的橋樑。

當然，現在來預測三十年後的情況似乎是太早一點了。就讓我們先來看看未來五年內，電腦科技的發展趨向。

當微型電腦出現之初，大家都未曾將它正式作一回事，只認爲是業餘發燒友攪出來的一種嗜好。對上一屆美國全國計算機會議（NCC）在美國舉行時，所有微型電腦、軟件及硬件全被冷落安排在一間酒店內的一個細小房間。令到支持會議的美國資料處理聯會處境非常尷尬。

可是，今年舉行的NCC大會，却全部是微型電腦的世界，電腦工業的老前輩，可能看不過眼，但他們却不能忽視這點。

過往的微型電腦純屬業餘發燒友的世界。可是現在經完全改觀。推動這潮流的其中一個重要人物，就是ADAM OSBORNE。雖然OSBORNE本人最近經驗嚴重的失敗，但由他一手設計的OSBORNE 1手提電腦在微型電腦發展史上佔著一個不可磨滅的地位。OSBORNE將大型電腦所做的工作，用一部手提的機器包裝起來，連同充足的軟件，以特惠的價錢發售。

另一個影響微型電腦發展的主流是BILL GODBOUT和他的COMPU－PRO 公司。他出售最先進的儀器。這些儀器被廣泛的利用在軟件的發展上。當然，你必須要具有微型電腦這方面的知識，方能充份發展COMPUPRO 的優點。

APPLE的LISA代表著第三條的發展主流。LISA稱不上是先進產品。若考慮它的內部裝置，可能更會嫌它太貴，但它完全是爲貪圖方便的人而設計。

可以說，GODBOUT和OSBO－RNE(或同類型）走著不同的發展路線。然而，却不能說它是代表了微型電腦未來的發展趨勢。微處理器將會應用在各種的家庭設備、汽車、電視機、家居保安系統，遊戲娛樂等方面。換言之，我們很難說出微型電腦的未來。

硬件

從硬件的發展趨勢來說，我們可以用一句說話來總括它的未來動向：較低的價錢得到更高的效能。

記憶體

記憶體的售價每年都在下跌！在EZEKIAL 推出之時，16K BYTE的高品質靜態記憶售價超過五百美元（以下同）。即每K爲31.25元。今日，第一級品質的靜態記憶每128 K售995元，即每K只售7.77 元。動態記憶體就更加可靠而且價錢更加廉宜。最近幾期BYTE雜誌廣告還說 256 K 只要 795元，即每K只花3.1元而矣。

新一代的記憶體將需要新的科技。現時將64K晶片一枚枚焊嵌在電路板上並肯定不能滿足需要。我們有理由相信新的技術會面世，不論在美國或日本。可以肯定說，五年後記憶體的價錢將不會超過現時之百分之十五。縱然是最微型的微型電腦也有一萬K的記憶量。

ROM

任何人都可預料，ROM的價錢亦一樣會下跌。ROM軟件將成爲電腦套件的一部份。機器本身將會有內置的操作系統、文字編輯等，而毋須依賴磁碟軟件。

大容量儲存器

筆者相信五年後，磁碟機依然使用，但只限於資料數據傳輸，而非用作大量儲存資料之用。一些工程師預測，未來五年內，8吋及5¼吋的磁碟將會被某種類型的硬碟取代，這些硬碟極有可能是盒式WINCHESTER。另一方面，假如3½吋袋裝型磁碟的標準規格能達成協議，將會令至5¼吋在市場消聲匿跡。

軟磁碟的售價，若扣除通貨膨脹，是在不斷的下跌中。硬碟方面更加戲劇性。五年前，一般人根本購不起硬碟，現時，16M BYTE連同控制卡的硬碟只售1595元，平均每K記憶量不需一美仙！

當然，除了硬碟外，尙有許多其他形式的大量儲存媒體選擇。例如：磁泡儲存器、電池儲存器，盒式磁碟等等。

我們毋須預測五年後會使用何種類型的儲存器，但可以肯定我們將會比現時付出更少，而獲得更多的儲存量。未來五年，一般的微型電腦記憶量起碼有10M BYTE。

中央處理器

那一種的矽晶片將會控制市場呢？目前拼過你死我活的是8086與68000。Z 8000 似乎稍爲遜色。而16032是隻黑馬。後者在結構上相當獨特。

衆多分析家均認爲8086的後來者

會有 2－86，而且相信會有更多後繼者。當然大家要拭目以待 68000 的動靜。至於要選擇那款，則完全視乎你的經濟能力。目前，這幾枚矽晶片經將VLSI的技術發揮至極限，要尋求突破無疑存有困難。假如再要求CPU表現能力加倍而價錢減半，則更難上加難。

接連

ETHERNET（美國網絡系統之一）現存最大的問題就是價錢昂貴。不過，一切將會成爲過去。原因並非由於它所需的硬件跌價（這個必然會成爲事實），而是連絡網絡與電腦之間的硬件將會獲得穩定的供應。

部份工程師，當RS－232C系統完全發揮其潛質時，其效能將會超乎我們所需。不過，部份持相反意見的工程師則堅持最快也要到1988年，方才有可靠及廉價的通信系統。筆者經常夢想有一日，安坐家中透過螢幕，交稿給電腦時代的編輯大人們。筆者只要用一支光筆，在螢幕上寫畫，便可跟遠處朋友通信，連撥電話也省却。目前限制這方面發展的主要因素，完全應由電話公司承担。

其他東西

MODEM 硬件、記憶管理系統、系統支援、數學常識、聲音辨別器等等均會在未來幾年內，價錢局部地減低。但對印字機用家來說就沒有如此幸運，因爲它的價錢變化不會太大。1978年時，一部有良好的打字機質素的印字機售3000元，現時同樣品質的只需 2000，到 1988.年或者可以用 1500元購得也說不定。

擊打式印字機需要高品質的金屬材料作爲構成機械部份，雖然在控制晶片方面成本可以減低，但整體來說，價錢不會相差太遠。

不過，兼具激光印字及複印功能的一部機器今年推出時一部售價爲15000元，而購一千部的價錢則爲每部 7000元。通常這些產品的售價趨勢是今年大量購買的價錢等於明年每一部的價錢。樂觀者認爲未來五年內，這部激光機器會打入辦公室，其售價不會超過普通影印機。

跌無止境的售價？

硬件的能力提高，但價格反而會下降，整個系統只會越趨便宜。

在 1977 年時，一部 Z80 的機器約售 12,000 元，包括軟件、系統綜合、打字機打字質素印字機、MODEM，電線及保養合約。以當時來說這部機器可算是非常先進。

可是今年的情形是：一部 ZEKE Ⅱ只要 8500 元，ZEKE Ⅱ的能力較 ZEKE Ⅰ強兩倍，速度快三倍。ZEKE Ⅱ有一個 1200 BPS的 MODEM 十倍記憶容量，有更多的軟件，而售價只是以前的百分之七十五。

照情形，這種趨勢一定會維持下去，大家不妨看看香港市面上現時有的機種，許多都較過去十二個月便宜。或者，五年後，你可以用現時家庭電腦的價錢，買到整套辦公室系統。當然，這並不將通貨膨脹計算在內。

軟件

可以預言，未來的軟件會劃一規格，各機適用，價錢更平，更容易使用。

EPSON AMERICA 公司最近在一個展覽會上展出一套新軟件：CHRIS RUTHOWSKI'S VALDOCS 系統，它用文字編輯作爲操作系統。事實上，這項安排非常正確，當開啓電腦時，它自動進入文字編輯狀態。假如要想與朋友聯絡，只需按兩個鍵便成，你可利用聲音（假如拿起電話筒），或通過電腦與電腦之間聯絡。用家又可以按下一個按鍵，編印出下週的約會。按另一個掣便變成計算機。

將來的應用軟件毋須再參閱繁複的用法說明書。軟件會自動告訴你。還有用聲音來控制系統的例如 APPLE 的 LISA。

它們都朝着一個方向發展，使很複雜的程序易於使用。務求使沒有興趣學習CP／M和BASIC的也可操作。

現時的軟件售價相當高昂（當然翻版軟件例外），主要原因是市場的有限。可是當銷售市場擴大開來時，一般價錢的訂價對軟件編寫員依然有利可圖。另方面有關的電腦參考書籍訂價雖然不高，但依然可以賺取利潤，養活著書者。未來軟件的情況亦差不多。

當微型電腦再進一步發展時，創造力和程式的設計比編寫電腦語言代碼的能力更爲重要，只要能使程式易於使用，管他用什麽語言來編寫！當記憶體越來越平時，實在不值得花錢請人編寫高效能電腦語言。幾年內，將會有許多軟件用LISP寫成，例如文字編輯、串字程式等。

目前來說，電腦在教育方面的功能還未被發展。專家們認爲，與其讓電腦教兒童工作，倒不如教兒童編寫電腦程式，教電腦工作。現時許多的程式化教學實在使人惋惜。透過電腦遊戲來學習目前只是起步，還有許多需要改善的地方。可以肯定說，教育軟件市場的潛力非常之龐大。雅達利最近亦宣佈將會投資相當大部份在教育軟件方面。由此可見，這將會是軟件發展的一條路向。

微型電腦有硬件的限制，但有並行處理，所以不成問題。數年前，有人預測「一個用家，一部CPU」。現在的多用家系統，與這預言剛好相反。

現在看來，我們不單祇「一個用家，用一部CPU」，還要一個用家，數部CPU。我們仍未忽視並行處理的用處。

現時所見的那種多用家系統，提供每個用家一些通過總綫接駁的CPU晶片，可以進行並行處理；大量的記憶體；一部終端機，幾種磁碟驅動器，而它可能只是一部小型WINCHESTER。它的操作系統可進行並行處理，所以它變成好像擁有幾部終端機，一部顯示文

字編輯，另一部顯示時間表和卡片文件箱，另一部接駁到一個"電子信箱"網絡。它又可以通過網絡接駁到另一用家，因而可共用一個激光印字機和一個大型硬磁碟。這系統與VAX的唯一分別，就是微型系統非常容易使用，而且價錢平。

明顯地，由於電腦趨向手提化和模塊化，所以當要寫新程序時，根本不需要重新開始寫。程序員只需帶一些模塊，將部份內容更改，以配合實際情況。

同時，我們可以看到一個趨勢，圖像越來越多樣化，適合不同的程序員。GRAPHICS 語句可以用很多種BASIC 語言來寫，例如OTRONA和ZENITH——100。這樣可幫助圖像廣泛使用在商業、教育和遊戲。

至於遊戲方面，VIDEODISCS，如PILOT的語言，和新的，平價的快速處理器可以開拓娛樂的新境界。相信不久便會實現。

直至現在，還未提及新的語言或類似PEARL 和 THE LAST ONE 的程式，這就是所謂程式編寫程式。用電腦來幫助編寫程式就是人工智能的其中一項成就。這將會接近邏輯的極限：即是說，只要你能說出程式要做的工作，你便能寫出這個程式。不過，五年來能否達到這成果，看來還需要長時間努力。

會說話的電腦

在這方面已經做了很多研究，可是這進展不會很快。肯定地，硬件在這方面大派用場，困難的是如何製造不碎的輭件，這將大部份依賴人工智能發展的情形，目前很難作出展望。

以下爲幾項預測：

遊戲程式將會在娛樂事業起很大作用。舊的公司如DOUBLEDAY和RANDOM HOUSE 等會出版這些遊戲的程式，在書局有售。

文字處理系統的銷量可比美打字機。

電話公司將會提供資料傳遞服務。除方便快捷外，還可提供各式各樣的資料服務，以適合各行各業的用家。

尾語

有些人會担心，可能有人會誤用這些資料而危害社會，這涉及複雜的法律問題，對律師來說，這是好消息，替他們帶來更多工作和收入！

發問的能力和如何找到資料亦很重要。五年內，電腦變成電視機一部份，到時，你買一部電視，你會得到一部電腦。還有就是所謂「電腦化家居」，人們可以在家裏工作，用電腦可聯絡，一星期才返回辦公室一次！■



最新 LOCKSMITH 5.0 版本面世

相信各位讀者之中，如有磁碟機，定然擁有最少一個抄錄程式工具，C—OPY A，NIBBLE AWAY，LOCKSMITH ……。不過道高一尺，魔高一丈，盡量許多專家設計出不少的保護方法，但終歸敵不住魔道中人，新的 COPY 程式不斷的陸續出現，設計 LOCKSMITH 的 OMEGA M—ICRONARE INC 不久前推出最新的 LOCKSMITH 5.0 版本，各位讀者又會多一件強勁的抄錄程式工具。

經過兩年不斷的研究修訂，LO—CKSMITH 5.0 版本終於能夠面世，讀者可以利用它來做任何磁碟的 BA—CKUP ，效果較任何 NIBBLE COPIER 更佳。

LOCKSMITH 5.0 版本在美國售 99.95 美元。有興趣的讀者不妨去函 OMEGA MICROWARE INC 查詢。該公司地址：

222 SOUTH RIVERSIDE
PLAZA，SUITE 2234，
CHICAGO, IL60606,
TEL：(312) 648—4844 ■

同步賽跑

本程式是一簡單的賽跑遊戲，可供二或三人玩。每人有四個以數字（1，2，3，4）代表的棋子，各以不同字體表示（分別爲 INVERSE，FLASH，NORMAL），遊戲的目的是要盡快將自己所有的棋子行到終點。

當遊戲開始，電腦便會顯視出跑道及棋子的位置（這時全部都在起點），各人可以輪流按 SPACE BAR 作爲擲骰，然後可以選擇移動其中一隻棋子（1，2，3，4）。

當你所行的棋子所停下的位置剛巧有別人的棋子，你可將它打回起點。如果是己方的棋子，則可重疊在一起。

當其中一方的所有棋子首先到達終點，遊戲便結束，同時電腦會顯示出勝利者。□

同步賽跑程式

```
100  TEXT : HOME : VTAB 2: HTAB 1
     6: INVERSE : PRINT "PATHWAY"
     : NORMAL
110  VTAB 6: HTAB 8: PRINT "TWO O
     R THERE PLAYERS? ";: GET A$:
     P =  VAL (A$): IF P < 2 OR P
      > 3 THEN 110
120  DIM A$(23),P(70),A(3,4):GV =
      - 16336: HOME : GOSUB 460
130  FOR I = 1 TO 70: READ P(I): NEXT

140 NM$(1) = "ONE":NM$(2) = "TWO"
    :NM$(3) = "THREE":OU(1) = 63
    :OU(2) = 127:OU(3) = 255
150  VTAB 2: HTAB 11: INVERSE : PRINT
     "1234": FLASH : HTAB 11: PRINT
     "1234": NORMAL : HTAB 11: PRINT
     "1234"
160  FOR I = 1 TO P: PRINT
170  VTAB 21: PRINT "     PRESS SP
     ACE": HTAB 7: PRINT "TO ROLL
     "
180  VTAB (5 * I - 3): HTAB 25: PRINT
     "PLAYER ";: POKE 50,OU(I): PRINT
     NM$(I): NORMAL : HTAB 25: PRINT
     "ROLLS:     ";: HTAB 34
190  GET A$: IF A$ <  > " " THEN
     190
200 V =  INT ( RND (1) * 9) + 1: FOR
    J = 1 TO V: HTAB 34: PRINT J
    ;: FOR K = 1 TO 200: NEXT : NEXT

210  PRINT : VTAB 21: PRINT "
     WHICH PIECE": HTAB 7: PRINT
     "TO MOVE"
220  VTAB (I * 5 - 1): HTAB 25: PRINT
     "MOVES:     ";: HTAB 34
230  GET A$:M =  VAL (A$): IF M <
     1 OR M > 4 THEN 230
240  IF A(I,M) > 70 THEN 230
250  IF A(I,M) = 0 THEN  POKE 103
     3 + 128 * I + M,160: GOTO 29
     0
270 QQ = 160: FOR K = 1 TO 4: IF
     A(I,K) = A(I,M) AND K <  > M
      THEN QQ = K + 64 * I - 16
280  NEXT : POKE P(A(I,M)),QQ
290 A(I,M) = A(I,M) + V: IF A(I,M
     ) > 70 THEN 350
300 A =  PEEK (P(A(I,M)))
310  IF A = 160 THEN 340
320  FOR J = 1 TO 3: FOR K = 1 TO
     4: IF A(J,K) = A(I,M) AND J <
      > I THEN A(J,K) = 0: POKE 1
     033 + 128 * J + K,A: PRINT  CHR$
     (7) CHR$ (7)
330  NEXT : NEXT
340  PRINT  CHR$ (7);: POKE P(A(I
     ,M)),M + 64 * I - 16: GOTO 4
     00
350  POKE 1385 + M + I * 128,I *
     64 + M - 16:C(I) = C(I) + 1:
      FOR K = 1 TO 25:AG = AG +  PEEK
     (GV): NEXT : IF C(I) < 4 THEN
     400
360  VTAB 17: HTAB 25: PRINT "PLA
     YER "NM$(I)"!": INVERSE : VTAB
     21: HTAB 5: PRINT "PRESS SPA
     CE": HTAB 4: PRINT "TO PLAY
     AGAIN";: NORMAL
370  VTAB 7: HTAB 32
380  GET A$: IF A$ <  > " " THEN
      HOME : END
390  RUN
400  NEXT : GOTO 160
410 :
420  DATA 1294,1295,1296,1297,129
     8,1299,1300,1428,1556,1684,1
     812,1811,1810,1809,1808,1807
     ,1806,1805,1804,1803,1802
430  DATA  1930,1074,1202,1203,12
     04,1205,1206,1207,1208,1209,
     1210,1211,1212,1340,1468,159
     6,1595,1594,1593,1592,1591,1
     590,1589,1588,1587,1586
440  DATA  1714,1842,1970,1971,19
     72,1973,1974,1975,1976,1977,
     1978,1979,1980,1124,1252,138
     0,1508,1636,1637,1638,1639,1
     640,1641
450 :
460 A$(1) = " ----"
470 A$(2) = "!    !-------"
480 A$(3) = "!              !"
490 A$(4) = "!    !----    !"
500 A$(5) = " ----       ! !"
510 A$(6) = " ----------    !"
520 A$(7) = "!              !"
530 A$(8) = "! !----------"
540 A$(9) = "! !----------"
550 A$(10) = A$(7)
560 A$(11) = " ----------! !"
570 A$(12) = A$(11)
580 A$(13) = A$(7):A$(14) = A$(8)
    :A$(15) = A$(9):A$(16) = A$(
    10):A$(17) = " ---------    !"

590 A$(18) = "                ! !"
600 A$(19) = "                ! !    --
    --"
610 A$(20) = "                !  ---!
    !"
620 A$(21) = "                !
    !"
630 A$(22) = "                  -----!
    !"
640 A$(23) = "                         --
    --"
650  FOR J = 1 TO 23: HTAB 10: PRINT
     A$(J): NEXT : RETURN
```

■

步步追踪

本程式是一個非常緊張刺激的追逐遊戲，名字稱爲步步追踪，莫因爲畫面構圖簡單，其難度之高超乎想像。玩者縱然自誇手腦靈活，最終也難逃一劫。

遊戲玩法相當簡單，在螢幕上（彩色螢幕），白點代表玩者，藍點及橙點（怪獸）則由電腦控制，玩者必須要避免給藍點或橙點追及，同時更不可以碰及洋紅色的圍牆，否則遊戲會自動結束。

每次當“怪獸”走過身旁而僥倖逃開便得一分。只要能碰紫色點，便可取得一千分獎分。不過假如被“怪獸”搶先一步，先接觸紫色得，獎分便會消失。

這個遊玩共有六級難度，由一至六，六是最高難度，在玩第二及第三難度時，只要注要不要闖進洋紅色的圍牆便可以了。到第四級難度時，可沒有那麼輕鬆了，因爲怪獸的移動速度跟白點一樣，而且怪獸們更佔有對角斜綫方面移動的優勢，第五級難度則是四處佈滿不能觸碰的洋紅色點，一旦不愼觸碰，遊戲便會結束。

第六級程度是最令人著迷的，圍牆內遍佈有九十個獎分點，每個獎分點有一千分獎分。白點可以自由走出牆外，只要使白點穿過牆中的獎分便可取得獎分了，不要担心白點會出了螢幕的界綫，遊戲內徑已設有一堵圍牆將螢幕圍繞！

本遊戲的控制如下：

→ 向右移動
← 向左移動
A 向上移動
Z 向下移動

□

步步追踪程式列表

```
90 M = 14
100  HOME
105  INVERSE
110  HTAB 15: PRINT "** CHASE **"
113  NORMAL
115  HTAB 13: PRINT "BY COMPUHOUS
     E LTD."
116  HTAB 15: PRINT
125  PRINT
130  PRINT "AVOID HITTING THE WAL
     LS, AND DO NOT LET THE MONST
     ERS CATCH YOU"
140  PRINT "USE THE ARROW KEYS TO
      MOVE LEFT AND      RIGHT, US
     E A AND Z TO MOVE UP AND DOW
     N"
150  PRINT : PRINT : HTAB M: PRINT
     "(1) NOVICE": HTAB M: PRINT
     "(2) ADVANCED ": HTAB M: PRINT
     "(3) MASTER": HTAB M: PRINT
     "(4) IMPOSSIBLE": HTAB M: PRINT
     "(5) RND DOTS": HTAB M: PRINT
     "(6) MULTI-SCORE"
160  PRINT : HTAB M: GET SK$:SK =
      VAL (SK$)
200  HTAB 8: VTAB 24: INPUT "PRES
     S RETURN TO CONTINUE";JJJ$
300  ONERR  GOTO 1050
500  GR : SPEED= 200
501  COLOR= 3: PLOT 32,22: PLOT 3
     1,12
510  COLOR= 1: VLIN 10,30 AT 10: HLIN
     10,15 AT 10: HLIN 17,28 AT 1
     0: VLIN 10,30 AT 30: HLIN 10
     ,30 AT 30
511  VLIN 7,10 AT 28: HLIN 28,32 AT
     7: VLIN 7,13 AT 32: HLIN 30,
     32 AT 13
512  IF SK = 5 THEN 514
513  GOTO 515
514  FOR I = 1 TO 40:X8 =  INT ( RND
     (1) * 27) + 7:Y8 =  INT ( RND
     (1) * 27) + 7: PLOT X8,Y8: NEXT
     I
515  VLIN 7,10 AT 17: HLIN 7,33 AT
     33: VLIN 20,33 AT 33: HLIN 3
     0,33 AT 20
```

```
517  IF SK = 6 THEN  HLIN 1,39 AT
     1: VLIN 1,39 AT 1: HLIN 1,39
      AT 39: VLIN 1,39 AT 39
520  HLIN 15,25 AT 15: HLIN 15,25
      AT 25: VLIN 15,25 AT 20: HLIN
     13,17 AT 20: HLIN 23,27 AT 2
     0: PLOT 20,26: PLOT 20,27: PLOT
     17,28: PLOT 17,29: PLOT 23,2
     8: PLOT 23,29
522  IF SK = 6 THEN 524
523  GOTO 530
524  COLOR= 3: FOR I = 1 TO 90:X8
      =  INT ( RND (1) * 37) + 2:
     Y8 =  INT ( RND (1) * 37) +
     2: PLOT X8,Y8: NEXT I: COLOR=
     1
530  REM  PLOR 5,16: REM PLOT 5,1
     7: REM PLOT 15,16: REM PLOT
     15,17
531  IF SK = 2 THEN  PLOT 8,18: PLOT
     9,21
532  IF SK = 3 OR SK = 4 THEN  PLOT
     8,18: PLOT 9,21: PLOT 8,24: PLOT
     9,27: PLOT 15,26: PLOT 15,27
     : PLOT 25,26: PLOT 25,27
540 OM = 1
550 MX = 29:MY = 29:P = 0: COLOR=
     7: PLOT MX,MY
560 X3 = 20:Y3 = 35:P = 0: REM  C
     OLOR= 7: REM PLOT X3,Y3
570 X2 = 15:Y2 = 20:P = 0: REM
        COLOR= 7: REM PLOR X2,
    Y2
580  HOME : VTAB 24: PRINT "USE A
     RROW KEYS TO MOVE DOT"
590 X = 22:Y = 20
600  COLOR= 15
610 KILL =  SCRN( X,Y): IF KILL =
     1 THEN  TEXT : HOME : PRINT
     "SCORE ";P: PRINT "YOU HIT T
     HE WALL!!": GOTO 1050
615  IF KILL = 3 THEN P = P + 100
     0: PRINT  CHR$ (7), CHR$ (7)

620  PLOT X,Y
630 ZZ =  PEEK ( - 16384)
640  REM  POKE - 16368,0
645 HAHA =  PEEK ( - 16336)
650  IF ZZ = 136 THEN  COLOR= 0: PLOT
     X,Y:X = X - 1
660  IF ZZ = 149 THEN  COLOR= 0: PLOT
     X,Y:X = X + 1
670  IF ZZ = 193 THEN  COLOR= 0: PLOT
     X,Y:Y = Y - 1
680  IF ZZ = 218 THEN  COLOR= 0: PLOT
     X,Y:Y = Y + 1
730 WM = WM + 1: IF WM = 3 THEN W
     M = 1
740  ON WM GOSUB 760,870
741  IF SK = 4 THEN  ON WM GOSUB
     870,760
750  GOTO 600
760  REM  ** MOVE MONSTER **
770 E2 =  SCRN( MX,MY): IF E2 = 1
      THEN  COLOR= 1
780  IF E2 < > 1 THEN  COLOR= 0
790  PLOT MX,MY: IF X > MX THEN M
     X = MX + 1
795 HAHA =  PEEK ( - 16336)
800  IF X < MX THEN MX = MX - 1
810  IF Y > MY THEN MY = MY + 1
820  IF Y < MY THEN MY = MY - 1
830  IF Y = MY AND X = MX THEN  TEXT
     : HOME : PRINT "THE MONSTER
     GOT YOU": PRINT : PRINT "SCO
     RE ":P: GOTO 1050
840 E1 =  SCRN( MX,MY): IF E1 = 1
      THEN  COLOR= 1
850  IF E1 <  > 1 THEN  COLOR= 7
860 P = P + 1: PRINT "SCORN ";P: PLOT
     MX,MY: RETURN
870  REM  ** MOVE MONSTER#2 **
880 E3 =  SCRN( X2,Y2): IF E3 = 1
      THEN  COLOR= 1
890  IF E3 <  > 1 THEN  COLOR= 0
900  PLOT X2,Y2: IF X > X2 THEN X
     2 = X2 + 1
905 HAHA =  PEEK ( - 16336)
910  IF X < X2 THEN X2 = X2 - 1
920  IF Y > Y2 THEN Y2 = Y2 + 1
930  IF Y < Y2 THEN Y2 = Y2 - 1
940  IF Y = Y2 AND X = X2 THEN  TEXT
     : HOME : PRINT "THE MONSTER
     GOT YOU": PRINT : PRINT "SCO
     RE ";P: GOTO 1050
950 E4 =  SCRN( X2,Y2): IF E4 = 1
      THEN  COLOR= 1
960  IF E4 <  > 1 THEN  COLOR= 9
970 P = P + 1: PRINT "SCORE ";P: PLOT
     X2,Y2: RETURN
1050  INPUT "DO YOU WANT TO PLAY
     AGAIN ";PL$
1060  IF  LEFT$ (PL$,1) = "Y" THEN
     100ELSE SPEED= 255: END
```

圍棋遊戲

跟電腦對奕

大家都會聽聞過圍棋，但相信懂得下圍棋的朋友不會很多吧，最主要原因是時間不容許，縱然懂得，但顯身手的機會不會很多，最終還是生疏拋掉。下圍棋必項要有充裕的時間，否則永不能完成終局。

筆者今次爲大家介紹一個相類似的遊戲，俾有圍棋天份而又未能遇到好對手的朋友一試身手。

遊戲的形式主要是在紙上劃上數排點，玩者輪流和電腦將點連接，每次兩點，不論垂直或水平也可。誰可以將四點連成正方形，便可得分，分數最高者，便是勝利者。

這個遊戲可用歡樂棒或鍵盤玩。玩者可在遊戲未開始前選擇“J”（歡樂棒）或“K”鍵盤控制輸入。當選擇完後，隨着是決定遊戲的難度，或0至10共十一個難度，10的難度最高。

遊戲開始時，螢幕會出現一個 10 點×10 點的遊戲盤。玩家先行，將黃色的游標（在彩色顯示中）移到希望連成一綫的兩點之間，按遊戲桿的發射器（或鍵盤的 RETURN 鍵）來輸入。電腦連接兩點時，會發出“必”聲，和閃動連成的綫，玩者可清楚看到移動的情況。

若連成一個正方形，可繼續多玩一次。在彩色顯示中，玩者連成的正方形會以象牙色顯示，而電腦則會用紅色顯示。在黑白或綠磷營幕中，兩者也很容易分辨。

當所有的 81 個正方形連成後，遊戲便終止。電腦會顯示最後的勝利者。□

圍棋遊戲程式列表

```
100  DIM XL%(23): FOR I = 0 TO 7:
     XL%(I) = 1024 + 128 * I:XL%(
     I + 8) = 1064 + 128 * I:XL%(
     I + 16) = 1104 + 128 * I: NEXT
     I
110  FOR I = 770 TO 795: READ M: POKE
     I,M: NEXT
120  TEXT : HOME : VTAB 11: HTAB
     19: INVERSE : PRINT "D O T S
     ": FOR J = 1 TO 1000: NEXT J
     : NORMAL
130  VTAB 15: HTAB 11: INPUT "SKI
     LL LEVEL (0-10) ?";SK: IF SK
     < 0 OR SK > 10 THEN 130
140 SK = (10 - SK) / 10:TS = 200 -
     200 * SK:DT = TS + 25
150  HOME : PRINT : PRINT : FOR I
     = 1 TO 10: PRINT : PRINT "
     0 0 0 0 0 0 0 0 0 0": NEXT
160  VTAB 9: HTAB 27: PRINT "USE
     KEYS:": VTAB 11: HTAB 27: PRINT
     "'J'=LEFT": HTAB 27: PRINT "
     'I'=UP": HTAB 27: PRINT "'K'
     =RIGHT": HTAB 27: PRINT "'M'
     =DOWN"
170  VTAB 7: HTAB 25: INVERSE : PRINT
     "            ": VTAB 16: HTAB
     25: PRINT "            "
180  FOR ROW = 7 TO 14: FOR COL =
     24 TO 36 STEP 12: POKE XL%(R
     OW) + COL,32: NEXT : NEXT : NORMAL
190 YS = 0:AS = 0: VTAB 1: HTAB 1
     3: PRINT "YOU: "YS"    APPLE
     :  "AS;
```

```
200  DEF  FN BX(COL) =  - ( PEEK
     (XL%(ROW + 1) + COL) <  > 16
     0) - ( PEEK (XL%(ROW) + COL +
     1) <  > 160) - ( PEEK (XL%(R
     OW - 1) + COL) <  > 160) - (
      PEEK (XL%(ROW) + COL - 1) <
      > 160)
210  DEF  FN BY(ROW) =  - ( PEEK
     (XL%(ROW + 1) + COL) <  > 16
     0) - ( PEEK (XL%(ROW) + COL +
     1) <  > 160) - ( PEEK (XL%(R
     OW - 1) + COL) <  > 160) - (
      PEEK (XL%(ROW) + COL - 1) <
      > 160)
220  DEF  FN VH(ROW) =  - ((XL%(R
     OW) + COL) <  > 2 *  INT ((X
     L%(ROW) + COL) / 2))
230 X = 10:Y = 10:ROW = 12:COL =
     10:CC =  PEEK (XL%(ROW) + CO
     L)
240  POKE XL%(ROW) + COL,106:F =
     0
250 OLDROW = ROW:OCOL = COL: GOSUB
     900
260 X = X + J:Y = Y + K: IF X < 1
      OR X > 19 OR Y < 1 OR Y > 1
     9 THEN X = X - J:Y = Y - K: GOTO
     250
270  IF PC = 99 THEN 300
280  POKE XL%(OLDROW) + OCOL,CC:C
     C =  PEEK (XL%(ROW) + COL): POKE
     XL%(ROW) + COL,42
290  GOTO 250
300 L =  - ( PEEK (XL%(ROW) + COL
      + 1) = 207) - ( PEEK (XL%(R
     OW) + COL - 1) = 207) - ( PEEK
     (XL%(ROW + 1) + COL) = 207) -
     ( PEEK (XL%(ROW - 1) + COL) =
     207)
310  IF L - (CC = 160) =  - 3 THEN
     330
320  POKE 768,250: POKE 769,1: CALL
     770: GOTO 250
330 WH = 1:MROW = ROW:MCOL = COL:
      GOSUB 750: IF  NOT  FN VH(R
     OW) THEN 380
340 BY = ROW: IF X > 1 AND  FN BX
     (COL - 1) =  - 4 THEN BX = C
     OL - 1: GOSUB 790:F =  - 1
350  IF X < 19 AND  FN BX(COL + 1
     ) =  - 4 THEN BX = COL + 1: GOSUB
     790: GOTO 230
360  IF F =  - 1 THEN 230
370  GOTO 410
380 BX = COL: IF Y > 1 AND  FN BY
     (ROW - 1) =  - 4 THEN BY = R
     OW - 1: GOSUB 790:F =  - 1
390  IF Y < 19 AND  FN BY(ROW + 1
     ) =  - 4 THEN BY = ROW + 1: GOSUB
     790: GOTO 230
400  IF F =  - 1 THEN 230
410 WH = 2:F = 0:CN = 0: IF  RND
     (1) < SK THEN 490
420  FOR ROW = 4 TO 22 STEP 2: FOR
     COL = 2 TO 18 STEP 2:K = XL%
     (ROW) + COL
430  IF  PEEK (K) = 160 AND  FN B
     X(COL) =  - 3 THEN 450
440  NEXT : NEXT : GOTO 490
450 I = K: IF  PEEK (XL%(ROW - 1)
      + COL) = 160 THEN ROW = ROW
      - 1: GOTO 640
460  IF  PEEK (XL%(ROW + 1) + COL
     ) = 160 THEN ROW = ROW + 1: GOTO
     640
470  IF  PEEK (XL%(ROW) + COL - 1
     ) <  > 160 THEN COL = COL +
     1: GOTO 700
480  IF  PEEK (XL%(ROW) + COL + 1
     ) <  > 160 THEN COL = COL -
     1: GOTO 700
490 ROW =  INT ( RND (1) * 19) +
     3:COL =  INT ( RND (1) * 19)
      + 1:CN = CN + 1: IF  PEEK (
     XL%(ROW) + COL) <  > 160 THEN
     490
500  IF  NOT (( PEEK (XL%(ROW) +
     COL + 1) = 207 AND  PEEK (XL
     %(ROW) + COL - 1) = 207) OR
     ( PEEK (XL%(ROW + 1) + COL) =
     207 AND  PEEK (XL%(ROW - 1) +
     COL) = 207)) THEN 490
510  IF SK > .6 OR CN > TS THEN 5
     70
520  IF  FN VH(ROW) =  - 1 THEN 5
     50
530  IF  FN BY(ROW - 1) =  - 2 OR
      FN BY(ROW + 1) =  - 2 THEN
     490
540  GOTO 640
550  IF  FN BX(COL - 1) =  - 2 OR
      FN BX(COL + 1) =  - 2 THEN
     490
560  GOTO 700
```

```
570  IF  FN VH(ROW) =  - 1 THEN 6
     10
580  IF SK > .6 OR CN > DT THEN 6
     40
590  IF  FN BY(ROW + 1) =  - 2 AND
      FN BY(ROW - 1) =  - 2 THEN
     490
600  GOTO 640
610  IF SK > .6 OR CN > DT THEN 7
     00
620  IF  FN BX(COL + 1) =  - 2 AND
      FN BX(COL - 1) =  - 2 THEN
     490
630  GOTO 700
640 MROW = ROW:MCOL = COL: GOSUB
     750
650 BX = COL: IF  FN BY(ROW - 1) =
      - 4 THEN BY = ROW - 1: GOSUB
     790:F =  - 1
660  IF  FN BY(ROW + 1) =  - 4 THEN
     BY = ROW + 1: GOSUB 790: GOTO
     410
670  IF F =  - 1 THEN 410
680  GOTO 230
690  IF ( PEEK (XL%(ROW - 1) + CO
     L) <  > 207 OR  PEEK (XL%(RO
     W + 1) + COL) <  > 207) THEN
     490
700 MROW = ROW:MCOL = COL: GOSUB
     750
710 BY = ROW: IF  FN BX(MCOL - 1)
      =  - 4 THEN BX = MCOL - 1: GOSUB
     790:F =  - 1
720  IF  FN BX(MCOL + 1) =  - 4 THEN
     BX = MCOL + 1: GOSUB 790: GOTO
     410
730  IF F =  - 1 THEN 410
740  GOTO 230
750  POKE 768,1: POKE 769,175: CALL
     770: IF WH = 2 THEN  FOR I =
     1 TO 500: NEXT I
760  POKE XL%(MROW) + MCOL,173: POKE
     XL%(MROW) + MCOL,45: FOR H =
     1 TO 50: NEXT H: POKE XL%(MR
     OW) + MCOL,173
770  IF  FN VH(MROW) =  - 1 THEN
      POKE XL%(MROW) + MCOL,201: POKE
     XL%(MROW) + MCOL,137: POKE X
     L%(MROW) + MCOL,201
780  RETURN
790 YS = YS + 1:J = 25:I = 200: IF
     WH = 2 THEN J = 1:I = 150:YS
      = YS - 1:AS = AS + 1
800  POKE 768,1: POKE 769,175: CALL
     770
810  POKE XL%(BY) + BX,J: VTAB 1:
      HTAB 13: PRINT "YOU: "YS"
      APPLE:  "AS;
820  IF YS + AS < 81 THEN  RETURN

830  IF YS > AS THEN 860
840  VTAB 24: HTAB 14: PRINT "SOR
     RY, YOU LOSE."
850  POKE 768,250: POKE 769,2: CALL
     770: FOR I = 1 TO 500: NEXT
     I: GOTO 880
860  VTAB 24: HTAB 15: PRINT "!!
     YOU WIN !!"
870  FOR I = 1 TO 5: POKE 768,1: POKE
     769,200 - I * 30: CALL 770: NEXT
     I: FOR I = 1 TO 10: POKE 768
     ,1: POKE 769,40 + I * 20: CALL
     770: NEXT I
880  VTAB 24: HTAB 14: PRINT "TRY
      AGAIN (Y/N) ?";: GET B$: IF
      LEFT$ (B$,1) = "Y" THEN 120

890  HOME : HTAB 5: VTAB 8: PRINT
     "...SEE YA...": END
900 PC = 0:J = 0:K = 0:A =  PEEK
     ( - 16384): IF A < 128 THEN
     900
910  POKE  - 16368,0:A$ =  CHR$ (
     A - 128)
920  IF A$ = "I" THEN ROW = ROW -
      SGN (ROW - 3):K =  - 1: RETURN

930  IF A$ = "K" THEN COL = COL +
      SGN (19 - COL):J = 1: RETURN

940  IF A$ = "M" THEN ROW = ROW +
      SGN (21 - ROW):K = 1: RETURN

950  IF A$ = "J" THEN COL = COL -
      SGN (COL - 1):J =  - 1: RETURN

960  IF A$ =  CHR$ (13) THEN PC =
     99: RETURN
970  RETURN
980  REM  MUSIC ML DATA
990  DATA  172,1,3,174,1,3,169,4,
     32,168,252,173,48,192,232,20
     8,253,136,208,239,206,0,3,20
     8,231,96
```

特惠訂閱

由即日開始訂閱電腦時代一年（十二期），可獲贈由CompuHouse製作之CompuHouse Disk* 一隻，只需付出港幣九十六元正（包括郵費）

同時訂購電腦時代＋CompuHcuse磁碟一年（十二期），只需付出港幣三百五十四元正（連郵費）

請填妥下列表格連同劃線支票寄回：

九龍中央郵政信箱71193號

電腦時代出版社收

*本刊每期刊出之所有程式，均由 CompuHouse 全部打入一張磁碟，省卻讀者自行打入程式的時間及可免卻捉錯打字的煩悶步驟。CompuHouse 製作之磁碟是採用高級貨，保證品質優越。

如要掛號，十二期加付港幣四十八元

（請用英文填寫，方便電腦處理）

NO. 4

本人欲訂閱

☑電腦時代一年（十二期）HK$96

☐電腦時代＋CompuHouse磁碟一年（十二期）HK$354

姓名：__________ 由第 4 期開始

地址：__________

電話：__________

附上劃線支票No. __________

支票收款人請填寫 "Computing Age Publisher"

o請寄九龍中央郵箱71193號

信封面請註明「訂閱」

香港電腦研習中心

Hong Kong Centre of Computer Studies

地址： 九龍旺角太子道141號長榮商業大廈506室
（即大大公司斜對面，地鐵太子站附近）

電話： 3-804040

本校特點：

一、師資優良：
全部講師爲香港大學畢業或以上程度。持有電腦或有關科目學士或碩士學位，並富有教學經驗。

二、充足實習時間：
置有十餘部電腦，供學員實習。星期一至六，上午十時至晚上八時，學員可隨時回校實習，不另收費。

三、備有教學軟件，學員可利用課餘時間，隨電腦學習。

四、講義由經驗講師精心編寫，深入淺出。

五、交通便利：
本中心位於旺角長榮商業大廈，即大大公司斜對面，地鐵太子站出口，恒生銀行樓上。歡迎參觀及查詢。（星期一至六，上午十時至晚上九時半）。

Database Management with dBASE II

包括用 dBASE II 處理資料及設計商用程式，全期學費連講義及實習共350元，逢星期六晚上課。

Advanced BASIC Programming

包括 programming and debugging techniques; CALL, PEEK, POKE; shape table, animated graphics; DOS; CP/M; data file programming; sorting and searching 等。全期學費連講義及實習共 320 元。

Introduction to Computer Programming

以教授BASIC電腦語言爲主，並包括電腦知識及磁碟操作系統（DOS）。全期學費連講義及實習共300元。

迎接電腦世紀的來臨
學習電腦語言與電腦溝通

北角國都電腦新樂園

李氏電子公司

大空型 JOYSTICK

打破傳統設計將兩個發射掣安裝在控制桿上，更易操作，可配合最新推出 JOYSTICK CARD 使用，
0-255 讀數可停留在任何位置，
AUTO FIRE,
電子控制，
非一般 VR 式可比。



ZETA SLIM COMPUTER DATA RECORDER

電腦專用錄音機
獨有 SAVE 及 LOAD 鑑聽和指示
AUTO STOP, 帶長 COUNTER
FOR APPLE COMPUTER
和多種電腦使用



" 香港區總代理，歡迎批發 "

SPEECH CARD II

中文，英文說話卡，更可幫你讀 CATALOG, DOS COMMAND, 外接式高品質小型喇叭箱，聲音響亮清晰 Software + Manual
只售弍佰餘圓

MUSIC CARD

40首世界名曲，及可以自己作曲

身歷聲輸出 Software + Manual

營業時間：上午 11:30 AM 至下午 9:30 PM

香港北角英皇道255號國都商場3樓213室
電話：5-718392
Shop No. 213 Olympia Shopping Centre,
2/F., 255 King's Rd., North Point, H.K.
TEL: 5-718392

香港灣仔道128-150號海運商場1樓1-2室
電話：5-8921792
Shop No. 1-2 Ocean Shopping Centre,
1/F., 128-150 Wanchai Rd., Hong Kong.
TEL: 5-8921792

COMPUTING AGE

電腦時代

第三期
一九八四年一月十五日
每月十五日出版
每本定價港幣八元正

編者話：
由於春節假期關係，電腦時代第四期將於二月二十日出版，不便之處，請讀者見諒。謹祝各位讀者
新春快樂

第三期程式磁碟同日出版
各特約經銷處有售　地址請參閱封底廣告

出　　版／電腦時代出版社
地　　址／英皇道 121 號嘉信大廈廿九樓A 4
編　　輯／電腦時代編輯部
通 訊 地 址／九龍中央郵箱71193號
通 訊 電 話／3-7776383
軟 件 支 援／CompuHouse
讀者服務部／深水埗南昌街白田購物中心二樓 197 室
承　　印／永恒印刷公司
葵涌工業街美安工業大廈12字樓 A 座
發　　行／同德書局
九龍西洋菜街 193 號

今期內容

▶程式儲存於磁碟上 所有程式均為APPLE版本

電腦人電腦事

一九八三年最佳電腦　5
花生門蘋果　7
港府全面推行中學電腦課程　10
TAXAN 產品推展會　6

特稿

WILDCARD 新版本 WILDCARD PLUS　9
COPYWRITER 卡介紹　10
門外漢選購印字機　12
LOCKSMITH 4. I PARAMETER　15
有十六種顯示功能的 ULTRATERM　21
未來五年微型電腦科技的發展　73
音樂產生的原理　32
▶巴哈古典名作　33
▶蘋果編曲器　35

工具程式

▶旋轉螢幕畫面　22
▶DOS 殺手　24
▶DOS 的自動同步追踪器　26

語言入門

機械語言程式輸入方法　11
PASCAL 與 APPLESOFT 的比較　28

應用程式

▶神經病醫生——神經鬆弛體操　46
▶製磁碟索引表封套　56
▶雜誌目錄分類歸檔系統　58

DOS 3.3 專欄

DOS BOOT 碟的原理　43

電腦教育·教育電腦

▶動量不滅定律　65
▶幼童認數字初階　68

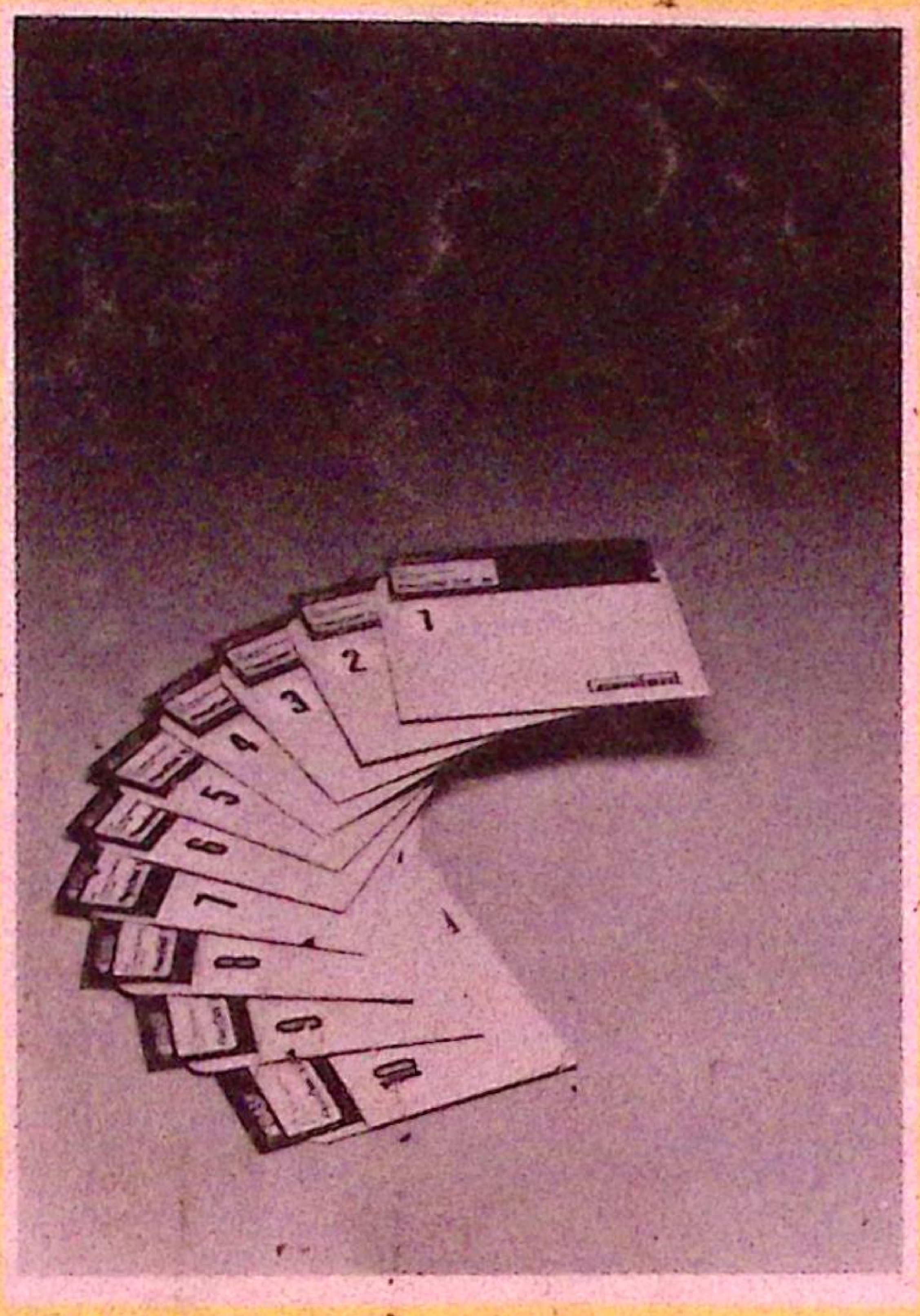
歡樂棒

▶賽跑遊戲　75
▶步步追踪　77
▶圍棋遊戲　79

其　他

LOCKSMITH 5.0 新版本面世　75
第二期巧破密令答案揭曉　64

電腦時代程式磁碟*



每一期電腦時代所登載的程式，均全部錄入一張磁碟同時出版，由COMPUHOUSE負責出版發行，各特約經銷處有售，每張售價港幣29.90元。

*All Programs are in Apple version

第3期程式磁碟內容：

DOS的自動同步追踪器	A DOS SELF-SYNCH DEMO
動量不滅定律模擬法	AIR CAR - MOMENTUM
自製磁碟索引表封套	APPLE DISKETTE ENVELOPE
蘋果編曲器	APPLE MUSIC WRITER
	PROGRAM#1-SOUND & VARIATION
	PROGRAM#2-PADDLE SOUND
	PROGRAM#3.OK
	PROGRAM#4.OK
	PROGRAM#5.OK
幼童認數字初階	BOUNCING NUMBER-MAIN PROGRAM
	MSOUND
	NUMBER TABLE
賽跑遊戲	PATHWAY
步步追踪	CHASE
DOS KILLER	DOS KILLER
圍棋遊戲	DOT
巴哈的古典樂章名作	FUNGUE
變轉螢幕畫面	LEAVING HOME
	LEAVING HOME DEMO
雜誌目錄分類歸檔系統	MAGAZINE FILE TRACKING SYSTEM
神經病醫生—神經鬆弛體操	RELAX ROUTS
	RELAX TEXT
	RELAXATION-MAIN PROGRAM
	CREATE RELAX FILES

以下為各特約經銷處地址：

百老滙沖印公司

Shop 1	**758A, Nathan Rd., Shop 8, G/F., Kowloon.**	**Tel: 3-816931**
Shop 2	**79, Broadway St., Stage IV, Mei Foo Sun Chuen, Kln.**	**Tel:3-7424923**
Shop 3	**746-8 Nathan Road, Kowloon.**	**Tel: 3-943827**
Shop 4	**3 Mut Wah St., Kwun Tong, Kowloon.**	**Tel: 3-419513**
Shop 5	**3, Queen Victoria St., Central, Hong Kong.**	**Tel: 5-235513**
Shop 7	**Shop 232 Cityplaza Taikoo Shing, Hong Kong.**	**Tel: 5-672006**
Shop 8	**207, Cheung Sha Wan Rd., Shamshuipo, Kowloon.**	**Tel: 3-796079**
Shop 9	**26, G/F., Soy Street, Mongkok, Kowloon.**	**Tel: 3-884331**
Shop 10	**60B, G/F., Argyle St., Mongkok, Kowloon.**	**Tel: 3-956367**
Shop 11	**230A, Sha Tsui Rd., Tsuen Wan, N.T.**	**Tel: 0-438926**
Shop 12	**35 G/F., Tung Yan St., Kwun Tong, Kowloon.**	**Tel: 3-449940**
Shop 13	**148 Ground Floor, Castle Peak Road, Kowloon.**	**Tel: 3-795938**
Shop 14	**529, Hennessy Road, Causeway Bay, H.K.**	**Tel: 5-770181**

亞當影音

ADAM CAMERA	**28A, Stanley Street, Central, Hong Kong.**	**Tel: 5-240723**
ADAM CAMERA	**Pioneer Building, G/F., 748A, Nathan Road, Kowloon.**	**Tel: 3-940526**

CompuHouse 地址：深水埗白田購物中心二樓197室　電話：3-7776383

讀者如欲補購第一期程式磁碟，可前往電腦時代讀者服務部/COMPUHOUSE購買

Apple is the trademark of Apple Computer Inc.